

UNCLASSIFIED

AD NUMBER
ADB108123
NEW LIMITATION CHANGE
TO Approved for public release, distribution unlimited
FROM Distribution authorized to U.S. Gov't. agencies only; Critical Technology; Mar 1986. Other requests shall be referred to Air Force Wright Aeronautical Laboratories, Attn: FIBRA, Wright-Patterson AFB, OH 45433.
AUTHORITY
WL/DOOS ltr, 24 Apr 1992

THIS PAGE IS UNCLASSIFIED

AD-B108 123

(2)

AFWAL-TR-86-3035

DESIGN GUIDE FOR BOLTED JOINTS IN COMPOSITE STRUCTURES

R. L. RAMKUMAR

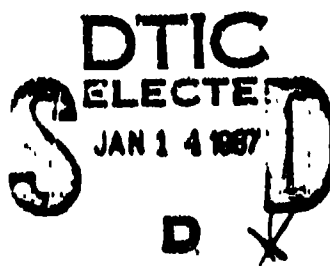
E. S. SAETHER

D. CHENG

Northrop Corporation, Aircraft Division
One Northrop Avenue
Hawthorne, California 90280

March 1986

Final Report for Period January 1983 to November 1985



Distribution limited to U.S. government agencies only,
test and evaluation, March 1986. Other requests
for this document must be referred to AFWAL/FIBRA,
WPAFB, OHIO 45433

WARNING—This document contains technical
data whose export is restricted by the Arms
Export Control Act (Title 22, U. S. C.,
Sec 2751 et seq.) or Executive Order 12470.
Violation of these export laws is subject
to severe criminal penalties.

DESTRUCTION NOTICE - Destroy by any
method that will prevent disclosure of
contents or reconstruction of the
document.

DTIC FILE COPY

FLIGHT DYNAMICS LABORATORY
AIR FORCE WRIGHT AERONAUTICAL LABORATORIES
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433

87 1 13 041

DISCLAIMER NOTICE

**THIS DOCUMENT IS BEST QUALITY
PRACTICABLE. THE COPY FURNISHED
TO DTIC CONTAINED A SIGNIFICANT
NUMBER OF PAGES WHICH DO NOT
REPRODUCE LEGIBLY.**

UNCLASSIFIED

ADB/08/23L

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1. REPORT SECURITY CLASSIFICATION UNCLASSIFIED		11. RESTRICTIVE MARKINGS T/E	
2. SECURITY CLASSIFICATION AUTHORITY		3. DISTRIBUTION/AVAILABILITY OF REPORT Distribution Limited to U.S. Government Agencies only; critical technology; 3.86. Other requests must be referred to AFWAL/FIBRA, WPAFB, OH 45433	
2a. DECLASSIFICATION/DOWNGRADING SCHEDULE		5. MONITORING ORGANIZATION REPORT NUMBER(S) AFWAL-TR-86-3035	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) NOR 86-211		7a. NAME OF MONITORING ORGANIZATION Flight Dynamics Laboratory (AFWAL/FIBRA) Air Force Wright Aeronautical Laboratories	
6a. NAME OF PERFORMING ORGANIZATION Northrop Corporation Aircraft Division	6b. OFFICE SYMBOL (If applicable)	7b. ADDRESS (City, State and ZIP Code) Wright-Patterson Air Force Base Dayton, OH 45433-6553	
8a. ADDRESS (City, State and ZIP Code) One Northrop Avenue Hawthorne, CA 90230		8. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER Contract F33615-82-C-3217	
9a. NAME OF FUNDING/SPONSORING ORGANIZATION	9b. OFFICE SYMBOL (If applicable)	10. SOURCE OF FUNDING NOS.	
8a. ADDRESS (City, State and ZIP Code)		PROGRAM ELEMENT NO. 62201F	PROJECT NO. 2401
11. TITLE (Include Security Classification) Design Guide for Bolted Joints in Composite Structures		TASK NO. 02	WORK UNIT NO. 55
12. PERSONAL AUTHOR(S) R. L. Ramkumar, E. S. Saether and D. Chang			
13a. TYPE OF REPORT Final	13b. TIME COVERED FROM 1/1/83 TO 1/30/85	14. DATE OF REPORT (Yr., Mo., Day) March 1986	15. PAGE COUNT 264
16. SUPPLEMENTARY NOTATION			
17. COSATI CODES		18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)	
FIELD	GROUP	Bolted joints, composite structures, design guidelines, analytical design tools (SASCJ and SAMCJ), user instructions, samples, program listing.	
01	03		
03	05		
19. ABSTRACT (Continue on reverse if necessary and identify by block number) A design guide was developed for bolted composite structural joints. The guide includes general design guidelines for the various joint parameters, an analytical design methodology, a description of the analytical design tools, an illustration of the use of corresponding computer codes (SASCJ and SAMCJ), and a listing of the computer codes. The proposed design procedure is purely analytical, and enables the user to rapidly evaluate many different joint concepts for a selected application. When the bolted structure is fabricated using existing (fully characterized) materials, the design requires no complementary test results. Presented analytical design tools are currently restricted to primarily uniaxially loaded joints and fastener arrangements that are currently used in aircraft structures. Also, the bolted joint is assumed to be strength-critical. However, sample fatigue test results are presented to illustrate a durability check on the joint, assuming a simplified fatigue analysis and assuming that fatigue failure is induced by excessive hole elongation. Despite its current restrictions, this guide is the first government document that provides guidance and analytical tools for the design of bolted composite structures.			
20. DISTRIBUTION/AVAILABILITY OF ABSTRACT UNCLASSIFIED/UNLIMITED <input type="checkbox"/> SAME AS RPT <input checked="" type="checkbox"/> DTIC USERS <input type="checkbox"/>		21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED	
22a. NAME OF RESPONSIBLE INDIVIDUAL V. B. Venkayya		22b. TELEPHONE NUMBER (Include Area Code) (513) 255-6992	22c. OFFICE SYMBOL AFWAL/FIBRA

DD FORM 1473, 83 APR

EDITION OF 1 JAN 73 IS OBSOLETE

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

PREFACE

This report was prepared under Contract F33615-82-C-3217, titled "Bolted Joints in Composite Structures: Design, Analysis and Verification," and administered by the Air Force Wright Aeronautical Laboratories. Dr. V. B. Venkayya was the Air Force project engineer, and was assisted by Capt. M. Sobota and Lt. D. L. Graves as program co-monitors. The program manager and principal investigator at Northrop was Dr. R. L. Ramkumar.

This report is a guide for the design of bolted joints in composite structures, and was prepared under Task 4 in the referenced program (Project 2401).

The authors gratefully acknowledge the support provided by the following personnel:

Graphics	R. Cordero
Typing	C. Harris



Accession For	
NTIS CRA&I	<input type="checkbox"/>
DTIC TAB	<input checked="" type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By	
Distribution /	
Availability Codes	
Dist	Avail and/or Special
B-3	51

TABLE OF CONTENTS

SECTION		PAGE
1	INTRODUCTION.....	1
	1.1 Scope of the Design Guide.....	2
	1.2 Sample Joint Configurations.....	2
	1.3 Overview of Design Methodology.....	2
	1.4 Analytical Requirements.....	12
	1.5 Test Requirements.....	12
	1.6 Current Restrictions.....	13
2	GENERAL DESIGN GUIDELINES AND JOINT VARIABLES.....	15
	2.1 Joint Location in the Structure.....	15
	2.2 Joint Configurations.....	17
	2.3 Joint Loading.....	17
	2.3.1 Joint Loads at the Structural Level.....	17
	2.3.2 Load Distribution Among Rows of Fasteners.....	20
	2.3.3 Bearing and By-Pass Loads at an Isolated Fastener Location.....	25
	2.4 Failure Modes in Bolted Laminates.....	27
	2.5 Fastener Type, Material and Installation Variables.....	27
	2.5.1 Fastener Type.....	27
	2.5.2 Fastener Material.....	29
	2.5.3 Fastener Size.....	30
	2.5.4 Fastener Fit and Hole Quality..	30
	2.5.5 Fastener Torque-up.....	33
	2.6 Bolted Laminate Properties.....	35
	2.7 Fastener Spacing and Arrangement.....	37
	2.8 Joint Tailoring for Maximum Efficiency.....	40
	2.9 Durability Considerations.....	46

TABLE OF CONTENTS (Concluded)

SECTION		PAGE
3	STRENGTH ANALYSIS OF BOLTED COMPOSITE STRUCTURES.....	49
	3.1 Description of SASCJ Analysis.....	49
	3.1.1 Strength Analysis Procedure for Fully-Loaded Holes.....	51
	3.1.2 Strength Analysis Procedure for Partially-Loaded Holes.....	53
	3.1.3 Inplane Failure Criteria.....	55
	3.1.4 Interlaminar Failure Criterion.....	61
	3.2 SASCJ Input Description.....	61
	3.3 SASCJ Output Description.....	68
	3.4 Description of SAMCJ Analysis.....	69
	3.5 SAMCJ Input Description.....	79
	3.6 SAMCJ Output Description.....	88
4	DESIGN VERIFICATION OF A BOLTED STRUCTURAL ELEMENT...	95
	4.1 Description of the Bolted Structural Element....	95
	4.2 Test Results.....	97
	4.3 Design Verification of the Element Using SAMCJ..	97
5	CONCLUSIONS.....	105
	REFERENCES.....	107
	Appendix A - SASCJ Program Listing.....	109
	Appendix B - SAMCJ Program Listing.....	183

LIST OF ILLUSTRATIONS

FIGURE		PAGE
1	Sample Bolted Joints in the F/A-18A Aircraft Wing (Reference 1).....	3
2	A Bolted Joint Concept for a Fuselage Structure (Reference 2).....	4
3	Bolted Joint Concepts for Composite Fuselage Structures (Reference 2).....	5
4	Skin-to-Root Fitting Joint in the F-20 Horizontal Stabilizer.....	6
5	Alternative Bolted Joint Concept Evaluated for the F/A-18A Wing Root Section (Reference 1).....	7
6	Alternative Bolted Joint Concepts Evaluated for the F/A-18A Vertical Tail Root Section (References 3 to 5).....	9
7	Overview of the Strength Analysis of Bolted Structures.....	11
8	Structural and Analyzed Bolted Joint Configurations..	18
9	Inplane Loads in Typical Wing Skin-to-Substructure Attachments.....	19
10	Sample Joint Configurations that Introduce Significant Out-of-Plane Loads at the Joint Location.....	21
11	Fastener Load Distribution in the Laminated Plate for Two Double Shear Configurations (References 7, 8)....	23
12	Effect of Joint Configuration on Fastener Load Distribution (Reference 9).....	24
13	Interaction Between Bearing and By-Pass Loads at a Fastener Location.....	26
14	Basic Failure Modes in Bolted Laminates and Fastener- Related Failures.....	28
15	Galvanic Compatibility and Corrosion Prevention.....	31

LIST OF ILLUSTRATIONS (Continued)

FIGURE		PAGE
16	Effect of Fastener Size on the Load Distribution (Reference 6).....	32
17	The Envelope of Bolted Laminate Layups for Realizing a Bearing Mode of Failure and the Maximum Bearing Strength (Reference 9).....	36
18	Geometrical Parameters for a Bolted Plate.....	38
19	Effect of Fastener Size on the Tensile Response of Composites-to-Metal Joints in Single Shear.....	39
20	Effect of E/D on the Bearing Strength of Bolted Laminates.....	41
21	Effect of W/D on the Bearing Strength of Bolted Laminates.....	42
22	Effect of S /D on the Strength of Bolted Laminates...	43
23	A Sample Tailored Joint.....	45
24	Effect of Maximum Cyclic Bearing Stress on the Number of R=-1 Fatigue Cycles to Cause Specified Hole Elongations in a Bolted Laminate.....	47
25	Effect of Maximum Cyclic Bearing Stress on Hole Elongation Rate for a Bolted Laminate Under R=-1 Loading.....	48
26	Schematic Representation of a General Single Fastener Situation as a Superposition of Unloaded and Fully- Loaded Hole Situations.....	50
27	Flowchart for the Strength Analysis of Laminates with Fully-loaded Holes.....	52
28	Strength Analysis of Laminates with Partially Loaded Holes Using Average Stress Failure Criteria...	54

LIST OF ILLUSTRATIONS (Continued)

FIGURE		PAGE
29	The Characteristic Distances Used in the Point Stress Failure Criteria.....	56
30	The Characteristic Distances Used in the Average Stress Failure Criteria.....	56
31	Bilinear Elastic Behavior of a Ply.....	58
32	A Schematic Representation of the Overall Load Versus Deflection Response of the Joint.....	59
33	The Characteristic Distance (a_0) Defining the Region Where the Maximum Strain, Hoffman or Hill Criterion is Applied.....	59
34	SASCJ Analysis of a Joint in Single Shear.....	63
35	SASCJ Analysis of a Joint in Double Shear.....	64
36	Sample SASCJ Input.....	65
37	SASCJ Output for the Problem Defined in Figure 36....	70
38	Flow Chart of SAMCJ Operations.....	72
39	Application of Load and Displacement Boundary Conditions in the SAMCJ Code.....	74
40	Finite Element Model of a Sample Tapered Bolted Joint.....	76
41	Element Load Recovery for Various a/D and b/D Ratios.....	78
42	Nine Element Model of Each of the two Bolted Plates in the Sample Joint.....	80
43	SAMCJ Input for the Sample Problem in Figure 42.....	81
44	SAMCJ Output for the Problem Defined in Figure 43....	89
45	Comparison of SAMCJ Predictions for the Sample Problem with Test Results from Reference 8.....	93
46	Photograph of an Assembled Test Element.....	96
47	Photograph of the Tab Region of the Failed Element.....	98

LIST OF ILLUSTRATIONS (Concluded)

FIGURE		PAGE
48	Dimensions of the Critical Skin Tab and the Fuselage Attachment Frame.....	99
49	Load Distribution Among Fasteners, Failure Location and Failure Mode in the Graphite/Epoxy Tabs.....	101
50	Analytically Predicted Element Load Levels to Precipitate Net Section, Bearing and Shear-Out Modes of Tab Failure at Each Fastener Location.....	103

LIST OF TABLES

TABLE		PAGE
1	GALVANIC COMPATIBILITY OF FASTENER MATERIALS WITH COMPOSITES (REFERENCE 12)	31
2	EFFECTS OF FLAWS AT FASTENER HOLE LOCATIONS (REFERENCE 13)	34

SECTION 1

INTRODUCTION

Bolted joints are a prime means of load transfer between structural parts in aircraft. Compared to other joining methods (bonding, welding, etc.), mechanical fastening is more reliable, with a potential for improved structural efficiency, maintainability and cost effectiveness. However, bolted joints are a source of stress concentration and could precipitate structural failures if they are designed improperly.

Prior to the initiation of this Northrop/AFWAL program, no analysis was available to be used as an exclusive design tool for bolted parts, especially if they were laminated composites. Consequently, their design has hitherto been based on extensive testing, empirical data and approximate analyses. The analysis developed in this Northrop/AFWAL program eliminates the need for extensive testing and provides a tool for the rapid evaluation of a bolted joint concept. If the structural part is to be fabricated using a characterized material, it eliminates the need for experimental information.

In the following sub-sections, the scope of this design guide is stated, sample bolted concepts are presented, criteria for the design of bolted joints in composite structures are discussed, the proposed design procedure is described, the analytical and experimental requirements for the design procedure are outlined, and its current restrictions are mentioned. In Section 2, general guidelines for the design of a bolted joint in composite structures are presented, along with summary statements on the effects of critical joint parameters. Section 3 presents the computer codes developed in this program for the strength analyses of single and multiple fastener joints in composites (SASCJ and SAMCJ, respectively). Section 4 demonstrates the use of the developed

analysis in predicting the strength of a realistic structural element.

1.1 Scope of the Design Guide

This design guide summarizes the effects of critical parameters on the strength and lifetime of bolted joints in composite structures, and presents general design guidelines. It also describes a test-independent analytical procedure for the strength evaluation of a bolted concept, based on the analyses developed in this program. The reader is familiarized with the computer codes (SASCJ and SAMCJ) that perform these analyses, and an application to a realistic structural bolted joint is demonstrated. This design guide will enable one to perform a rapid analytical evaluation of many joint configurations, and to select an efficient bolting concept. The described computer codes are currently restricted to uniaxial loading, conventionally used fastener spacing and protruding head fasteners.

1.2 Sample Joint Configurations

Figure 1 presents six composite-to-metal bolted joint configurations used in the F/A-18A aircraft wing (Reference 1). Figures 2 and 3 present joint configurations used in a typical fuselage structure (Reference 2). A skin-to-root fitting bolted joint in the F-20 horizontal stabilizer is shown in Figure 4. Many bolted joint concepts have been studied recently as potential alternative joining concepts for the F/A-18A wing root section and the F/A-18A vertical tail root section (Figures 5 and 6, respectively). The sample bolted configurations in Figures 1 to 6 illustrate the possible variety in this joining concept.

1.3 Overview of Design Methodology

There are many variables in the design of a bolted joint in composite structures. These include the geometry and the

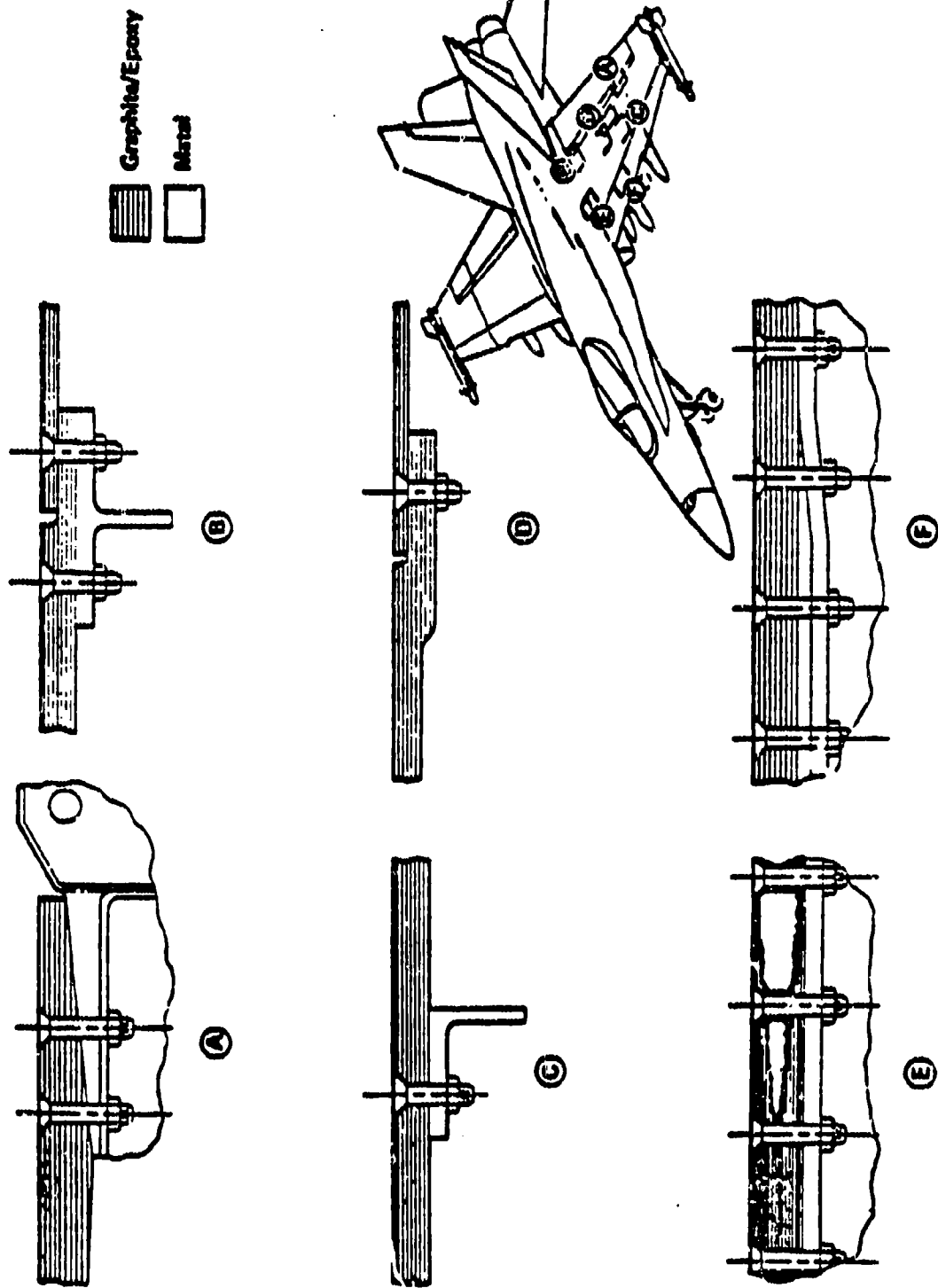
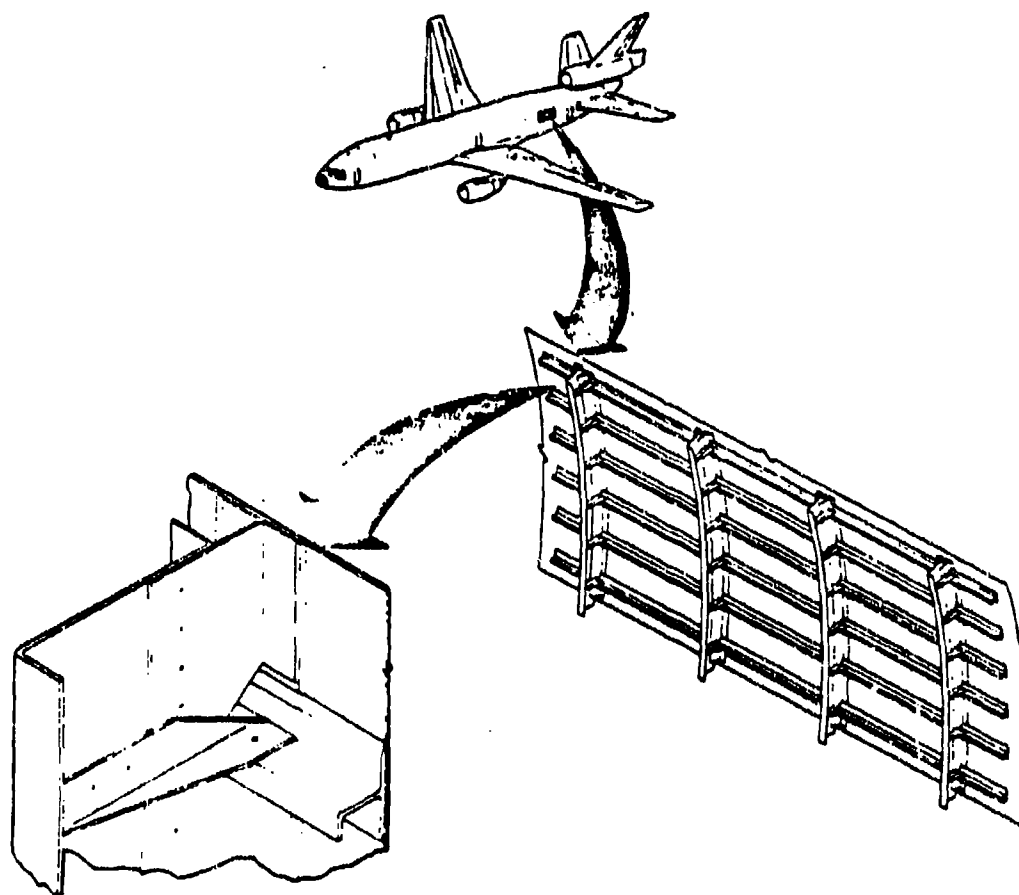


Figure 1. Sample Bolted Joints in the F/A-18A Aircraft Wing (Reference 1).



STIFFENER ATTACHMENT

Figure 2. A Bolted Joint Concept for a Fuselage Structure
(Reference 2).

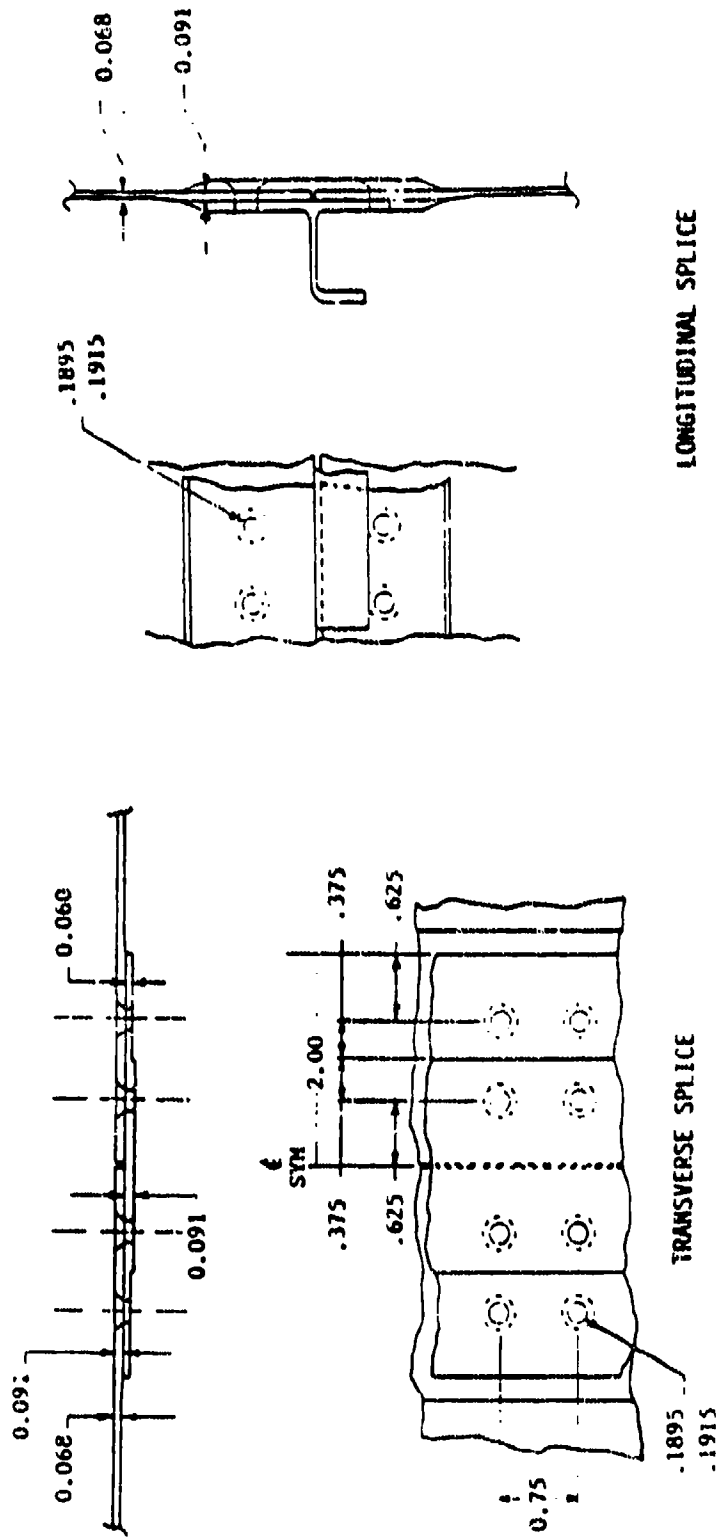


Figure 3. Bolted Joint Concepts for Composite Fuselage Structures (Reference 2).

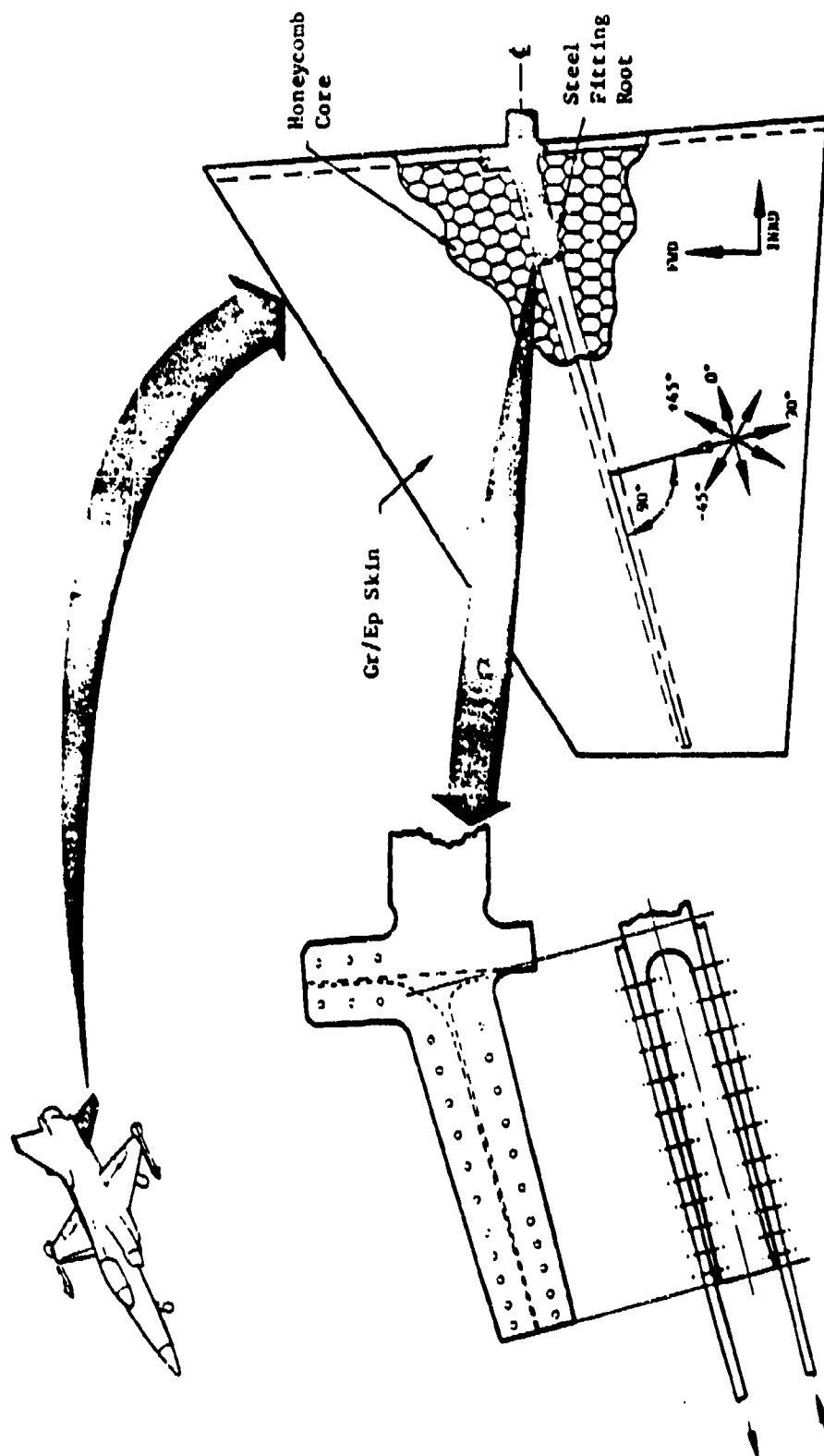


Figure 4. Skin-to-Root Fitting Joint in the F-20 Horizontal Stabilizer.

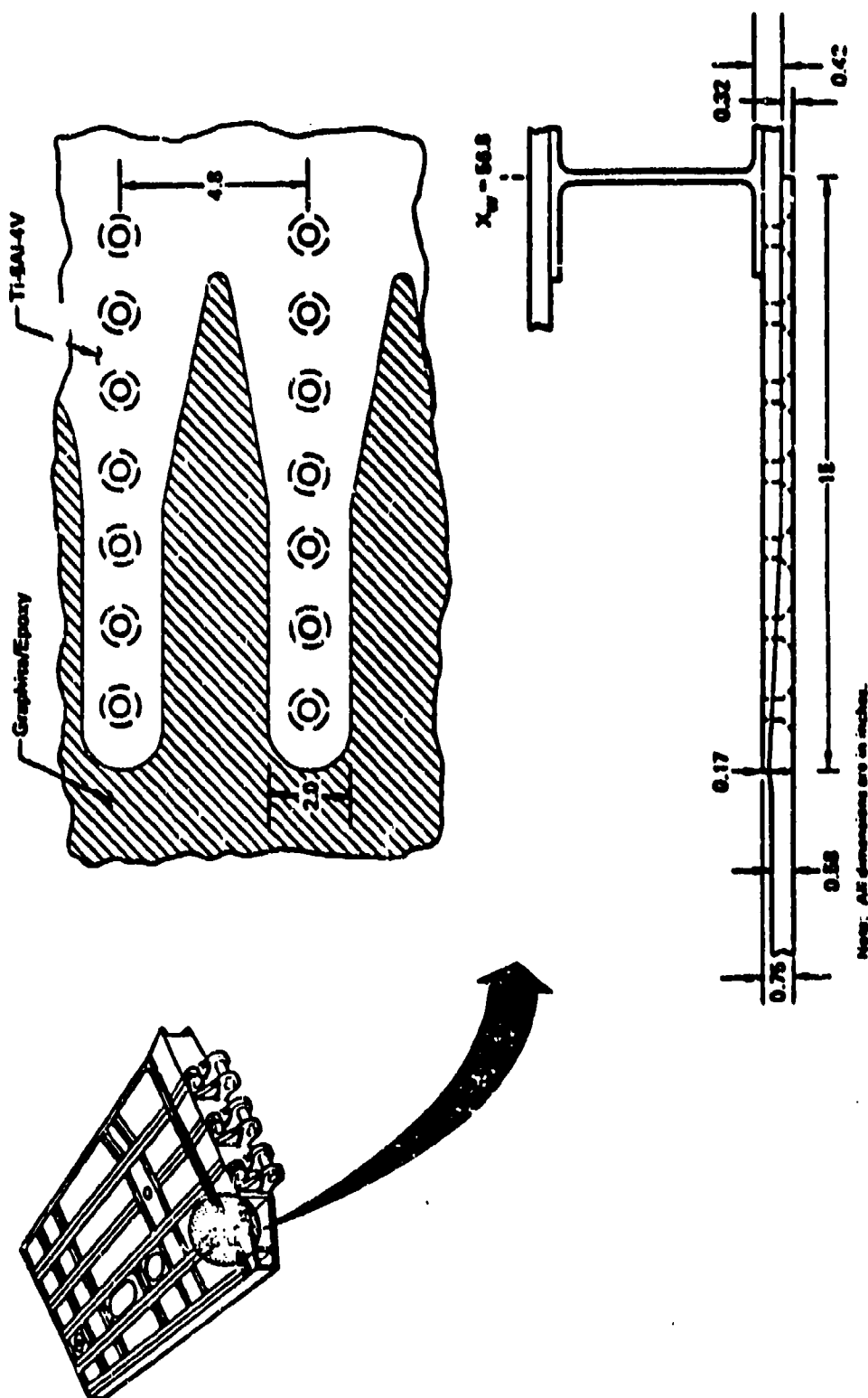
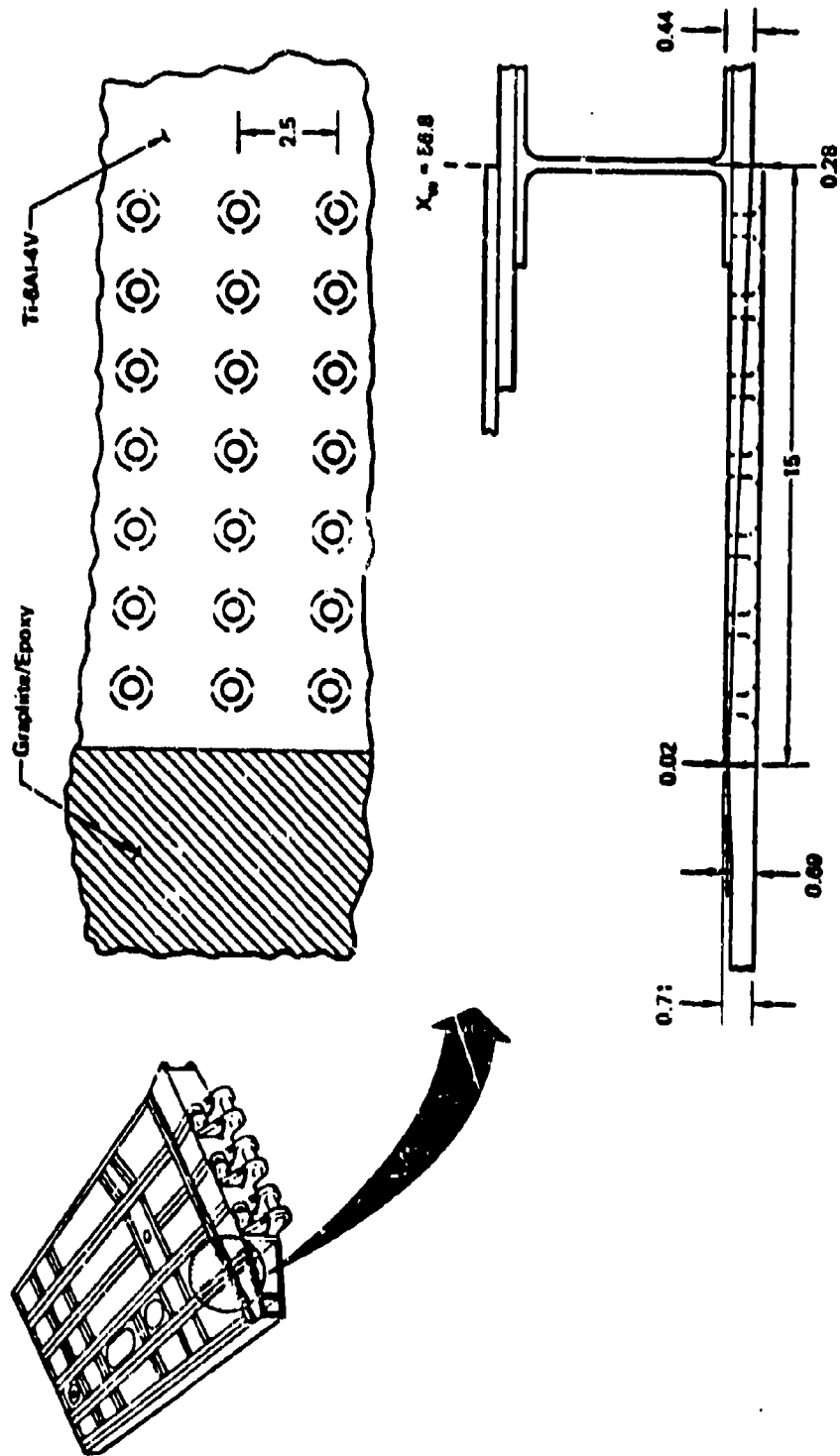


Figure 5 . Alternative Bolted Joint Concept Evaluated for the F/A-18A Wing Root Section (Reference 1).



Note: All dimensions are in inches

Figure 5. Alternative Bolted Joint Concept Evaluated for the F/A-18A Wing Root Section. (Reference 1; Concluded).

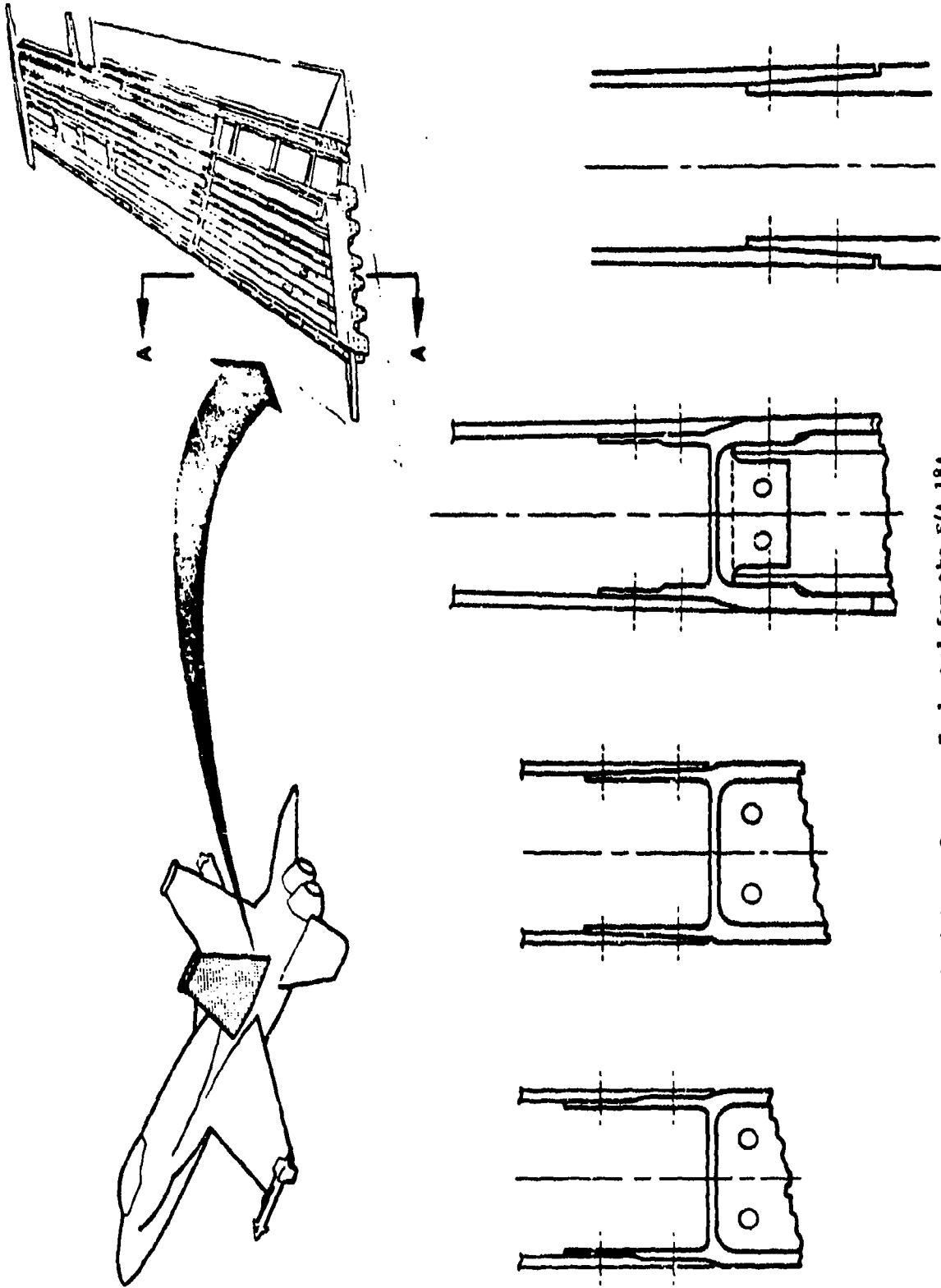


Figure 6. Alternative Bolted Joint Concepts Evaluated for the F/A-18A Vertical Tail Root Section (References 3 to 5).

material properties of the bolted parts, the size and arrangement of the fasteners, the fastener material properties and torque, applied loading and the load transfer mechanism (single versus double shear), etc. The design of a bolted joint involves a parametric study of the effects of the above variables on the joint efficiency, for a specified loading condition. A preliminary analysis of a structural component, based on conventional assumptions, yields the general biaxial loading transferred at the joint location (see Figure 7). The design procedure recommended in this guide assumes a predominantly uniaxial loading at the joint location.

The design of a uniaxially loaded joint in composite structures may be performed using the analyses developed in this Northrop/AFWAL program. Section 3 describes the use of the SASCJ and the SAMCJ computer codes for the strength prediction of single and multiple fastener joints in composites, respectively. The SASCJ code predicts the strength of joints when a single fastener transfers the applied load between the bolted plates. This analysis accounts for material nonlinearity in the bolted plates, the non-uniform fastener load distribution in the thickness direction of the bolted plates, and the progression of ply-level failures based on a choice among a few failure criteria. The SAMCJ code predicts the strength of plates bolted together by one or many fasteners. It computes the magnitude and orientation of the load at every fastener location, the applied load level for averaged stress components to reach critical levels at fastener and cut-out locations, the failure value of the applied load, the failure location and the failure mode (net section, shear-out or bearing). Failure predictions are made at the laminate level using average stress failure criteria.

The proposed design procedure involves the use of the developed analyses to evaluate the effect of joint variables on joint efficiency. If the bolted plates are fabricated using characterized materials, the joint design is tested-independent.

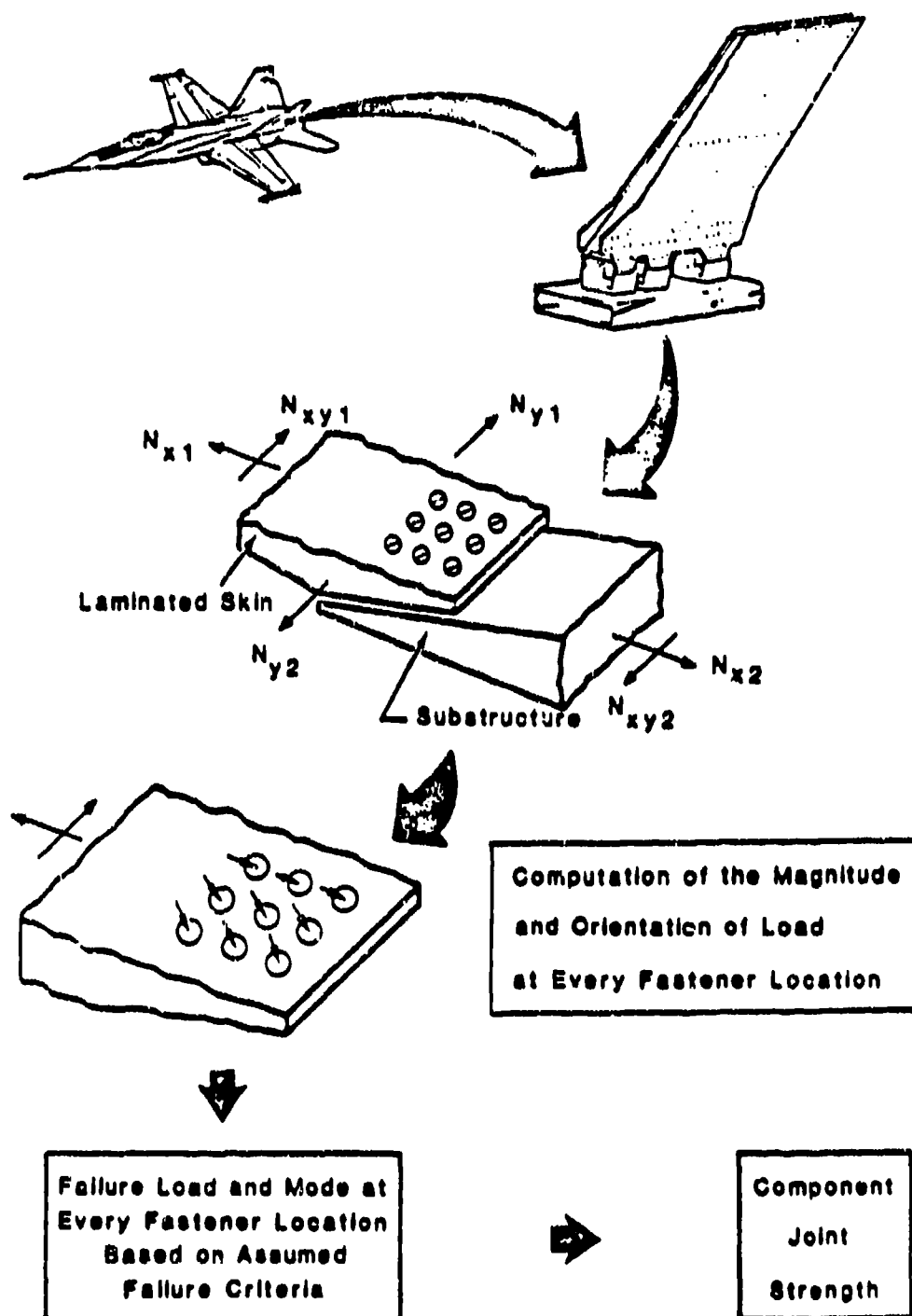


Figure 7. Overview of the Strength Analysis of Bolted Structures.

Candidate bolted joint concepts are selected following the general guidelines outlined in Section 2. The fastener size and arrangement (spacing between fasteners), the geometry of the bolted plates, the load transfer mechanism, etc. are varied without violating the constraints imposed by the structural application. The strength and durability of each bolted joint concept, along with its impact on manufacturing costs and maintenance, are evaluated to establish joint efficiency. An efficient bolted joint concept can thus be designed using a purely analytical tool on a finite number of concepts that are selected in accordance with established guidelines.

1.4 Analytical Requirements

The design of a bolted joint for composite structures requires the analyses developed in this Northrop/AFWAL program (References 6 and 7). The analysis of plates bolted together by a single fastener may be performed using the SASCF (Strength Analysis of Single Fastener Composite Joints) or the SAMCF (Strength Analysis of Multiple Fastener Composite Joints) computer code. Plates bolted together by many fasteners are analyzed using SAMCF computer code. Section 3 presents a brief description of these analyses. The reader is referred to References 6 and 7 for further details.

1.5 Test Requirements

A test-independent, purely analytical design tool has been developed to design a bolted joint for composite structures that are fabricated using characterized materials. The engineering properties (Young's moduli in the fiber and transverse directions, major Poisson's ratio and the shear modulus in the fiber coordinate system), the strengths or failure strains (under tension, compression and shear), and the failure parameters for the assumed failure criteria (characteristic distances for net section, shear-out and bearing failure predictions using the average stress

failure criteria, for example) are known for a characterized composite material (lamina). Tests required to obtain the above material properties must be performed on a new (uncharacterized) material system, prior to designing bolted joints for structural parts made from this material. When previously characterized materials are used in the bolted plates, the test requirements are nil for the design of an efficient bolted joint concept.

1.6 Current Restrictions

The design of bolted joints in composite structures is influenced by the current restrictions in the developed analytical tools. The primary restrictions are listed below:

- (1) The developed strength analyses (SASCJ and SAMCJ computer codes) do not account for countersunk fastener effects.
- (2) SASCJ and SAMCJ contain a stress analysis that approximates the fastener/plate contact problem by an assumed radial stress distribution.
- (3) SASCJ AND SAMCJ are restricted to a uniaxial applied loading, in tension or in compression.
- (4) The prediction of the durability of a joint is restricted to the incorporation of the bearing stress at critical fastener locations into experimentally obtained curves for joint life.
- (5) SAMCJ restricts the user to rectangular element geometries and currently used fastener spacing and arrangement.

Despite the above restrictions, the developed analyses and the proposed design procedure mark a significant improvement

over the state-of-the-art with respect to the design and analysis of bolted joints in composite structures.

SECTION 2

GENERAL DESIGN GUIDELINES AND JOINT VARIABLES

The design of boted joints in composite structures involves the definition of many variables. The major design considerations are listed below:

- (a) The loads that must be transferred from one part to another.
- (b) The load transfer location in the structure.
- (c) Geometric constraints, if any, at the load transfer location.
- (d) Fastener type, size and arrangement.
- (e) The environmental range the joint will be exposed to.
- (f) The effect of the joint concept on structural efficiency and reliability.

The following sub-sections discuss the primary variables that influence the design of boted joints in composite structures. Design guidelines corresponding to the discussed joint parameters are highlighted within the sub-sections.

2.1 Joint Location in the Structure

The location of the joint in a structure influences the selection of the joint variables significantly. Design guidelines pertaining to selected joint locations are presented below:

- (a) When aerodynamic surfaces in an aircraft structure are joined to substructural parts, or segments of a surface are joined together, the requirement of a smooth outer moldline should not be

violated. The use of protruding head fasteners on such surfaces, or the presence of any other geometric discontinuity (step) at the joint location, will adversely affect the lift distribution on these surfaces and their aerodynamic performance.

On aerodynamic surfaces, fasteners must be installed to be flush with the surface, without exposed fastener heads, and joined members must retain a smooth outer moldline.

(1)

(b) When structural members are joined together in fuel-containment areas, measures must be taken to preclude leakage of the fuel and service-related hazards. The use of metallic fasteners on the outer surface, for instance, introduce the threat of arcing within the fuel cell in the event of a lightning strike. In designing joints for these locations, special consideration must be given to the mentioned sealing requirements.

In fuel containment areas, joints must be sealed to be leak-proof. Fasteners must also be sealed to prevent arcing within the fuel cell in the event of a lightning strike.

(2)

(c) When bolted joints are designed for structural regions with limited or restricted access, special fastener types have to be used.

In areas of restricted accessibility, blind fasteners must be used.

(3)

(d) When a laminated part is bolted to a metallic substructure, the threat of joint corrosion must be considered.

In composite-to-metal joint locations, corrosion barriers like fiberglass layers must be used.

(4)

2.2 Joint Configurations

Selected joint configurations are significantly influenced by their structural locations. Figures 1 to 6 present typical structural joint configurations in current aircraft. Figure 8 presents the localized structural joint configurations along with their equivalent configurations that are analyzed. The configurations that transfer loads in single shear introduce localized bending effects that could adversely affect the strength and durability of the joint. Stepped lap and scarf configurations involve thickness changes that provide an additional design variable (layup) in bolted laminates.

2.3 Joint Loading

Structural joints are designed to be effective over their design lifetime, when subjected to the anticipated design spectrum fatigue loading. The durability considerations for structural joints are discussed in Section 4. This design guide emphasizes the strength analysis of a bolted joint, and presents computer codes that perform it. The reader must supplement the joint design based on a strength analysis with a durability check, using information similar to that presented in Section 2.9. The effect of joint loading is discussed further below, at three levels -- structural, among fastener rows, and at an isolated fastener location.

2.3.1 Joint Loads at the Structural Level

Joint loads at the structural level fall into two basic categories -- inplane loads and out-of-plane or bending loads. Figure 9 presents some possible inplane load conditions in typical wing skin-to-substructure attachments. The analyses developed in this Northrop/AFWAL program, and described in Section 3, assume that the joint at each location is subjected to a predominant unidirectional load. Figure 9 illustrates that this assumption will

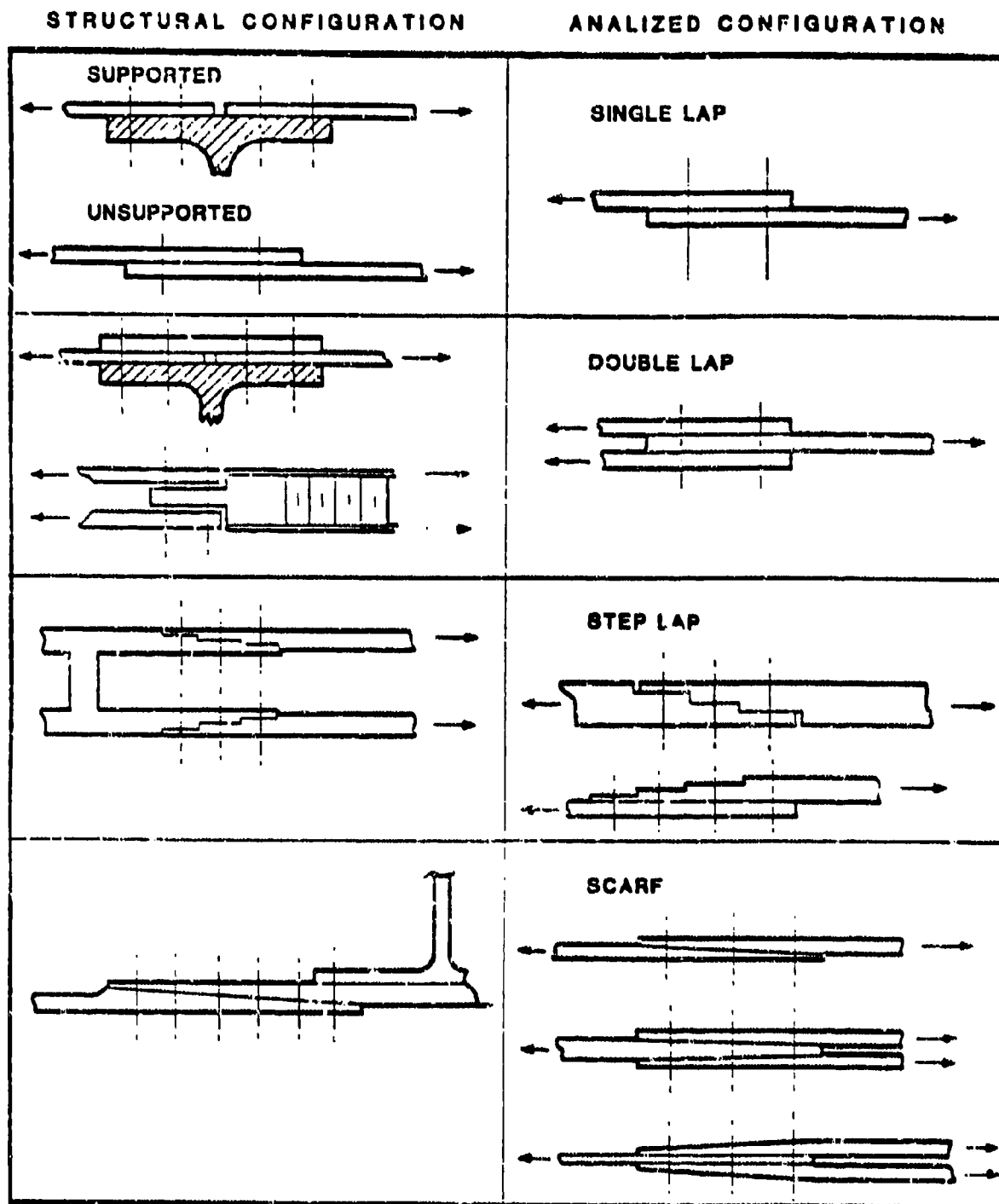


Figure 8. Structural and Analyzed Belted Joint Configurations.

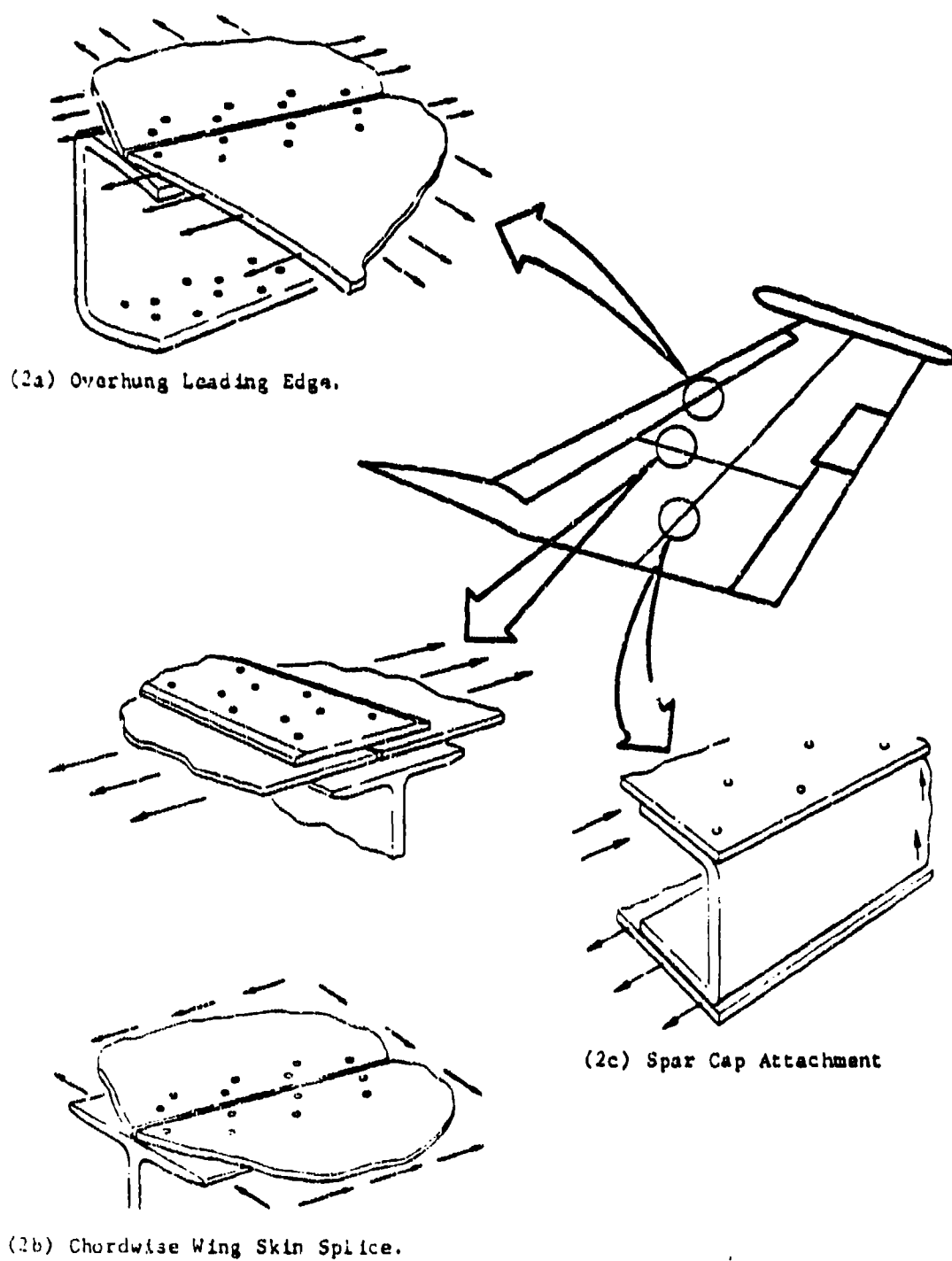


Figure 9. Inplane Loads in Typical Wing Skin-to-Substructure Attachments.

not be valid at some locations.

Figure 10 presents sample situations where considerable out-of-plane (bending) loads are introduced at the joint location. This is inherent in single shear load transfer configurations (see Figure 8), and adversely affects joint strength and durability. If one of the bolted plates is very stiff compared to the other, the deleterious effects of load eccentricity in a single shear configuration are minimized. In double-shear load transfer configurations (see Figure 8), the out-of-plane loads are reduced to a negligible level.

Single - shear load transfer joint configurations introduce out-of-plane (bending) loads that could significantly reduce the strength of the joint. When one of the bolted members is very stiff, the effect of the out-of-plane loads is minimized.

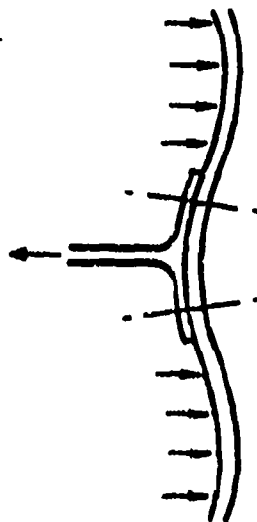
(5)

Double - shear load transfer joint configurations essentially introduce inplane loads in the bolted plates.

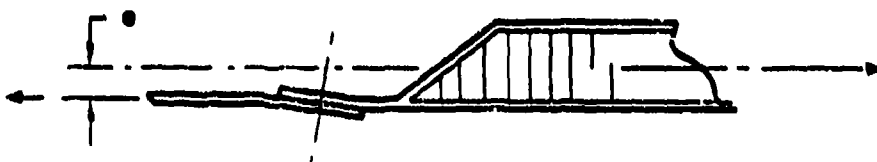
(6)

2.3.2 Load Distribution Among Rows of Fasteners

Assuming a unidirectional applied load, the fasteners in a row are arranged perpendicular to the load direction. Joint configurations affect the distribution of the applied load among the various rows of fasteners in a joint, and the distribution of the row-wise load fraction among the fasteners in any row. Hitherto, the fasteners in a row have been assumed to carry equal loads, and only the row-wise load distribution has been analytically predicted. The SAMCJ code developed in this Northrop/AFWAL program overcomes this limitation, and predicts the two-dimensional load distribution (magnitude and orientation of fastener loads at all locations) for a selected fastener pattern.



**A. OUT-OF-PLANE JOINT LOADING DUE TO INTERNAL PRESSURE
(e.g., FUEL PRESSURE, FUSELAGE CABIN PRESSURE, ETC.)**



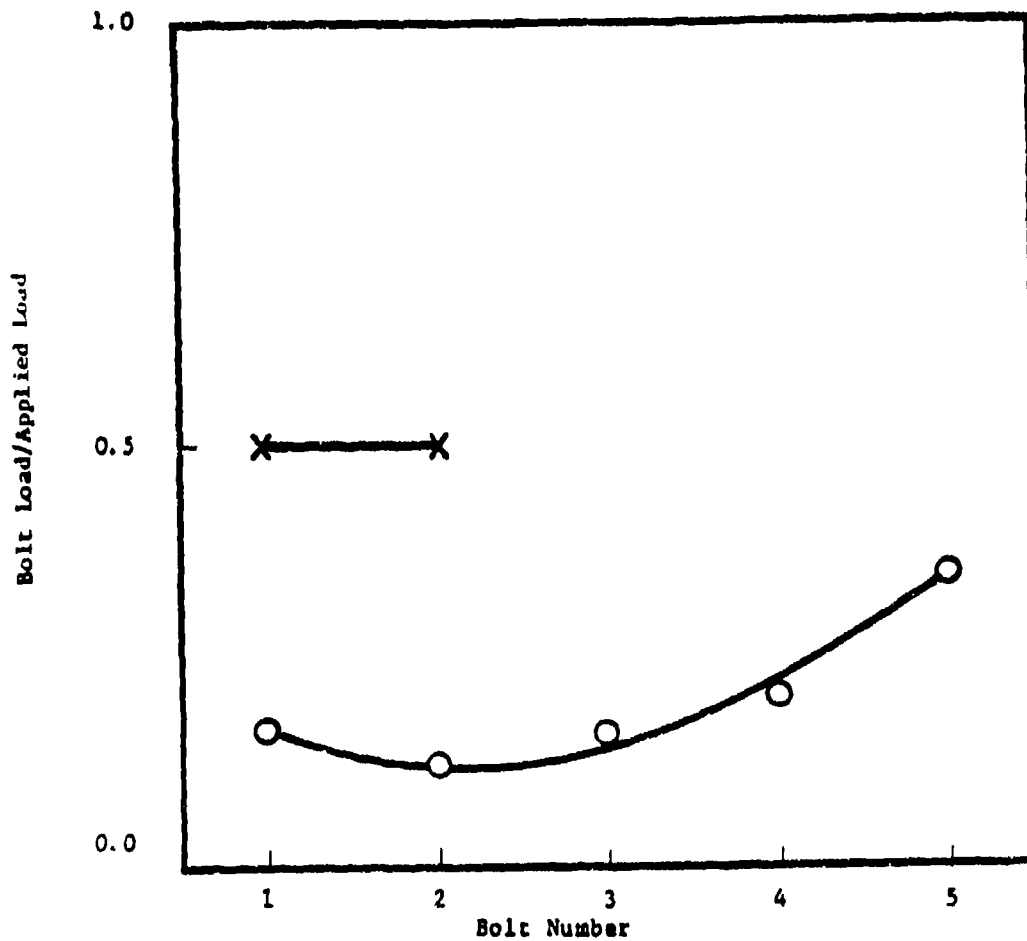
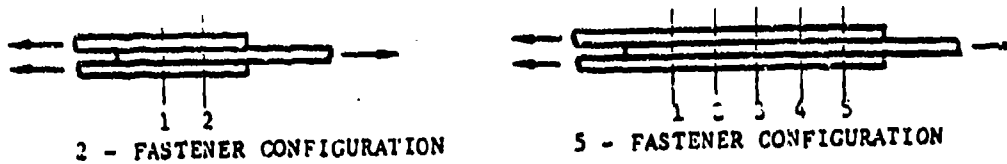
B. OUT-OF-PLANE JOINT LOAD DUE TO LOAD PATH ECCENTRICITY

Figure 10. Sample Joint Configurations that Introduce Significant Out-of-Plane Loads at the Joint Location.

Figure 11 presents the load distributions for two and five fastener, double shear joint configurations tested in this program (References 7 and 8). The bolted plates in Figure 11 were uniform in thickness. Figure 12 illustrates how the load distribution among four rows of fasteners can be varied by changing the joint configuration. In the strongest configuration (4), a combination of tapering and reinforcing of the splice plates minimizes the bearing load where the by-pass load is the largest (station 1), and maximizes the bearing load where there is no by-pass load (station 4). The plate width-to-bolt diameter ratio (W/D) is 5 at station 1, and 4 at stations 2 and 3. A larger bolt is used at station 4 ($W/D=3$). This results in a reduction of the bearing stresses at stations 2 to 4, and the strongest configuration (see References 9 and 10).

In bolted metallic plates, the fastener load distribution is similar to those shown in Figure 11 for low values of the applied load. But, as the applied joint load increases toward the failure value, yielding will occur at peak fastener load locations. This causes the incremental applied load to be carried by the remaining fasteners, generally resulting in a uniform fastener load distribution near failure. For the five fastener configuration in Figure 11, for example, every fastener will carry one-fifth of the applied load at failure. However, laminated plates generally exhibit a linear elastic and brittle behavior, with negligible ductility or yielding. The non-uniform load distribution among rows of fasteners in composite laminates, therefore, remains non-uniform at failure. This reduces the failure load level if the peaks in the load distribution are not accompanied by appropriate thickness tapering and other changes in the joint configuration. Joint efficiency is determined by the overall load-carrying capability of the joint.

The load distribution among rows of fasteners in a bolted laminate generally remains non-uniform at the failure load level, in contrast to what is



Note: Double-shear load transfer between 50/40/10, AS1/3501-6 graphite/epoxy laminate and aluminum using 5/16-inch diameter, protruding head steel fasteners torqued to 100 in-lb ; static tension; RTD.

Figure 11. Fastener Load Distribution in the Laminated Plate for Two Double-Shear Configurations (References 7, 8).

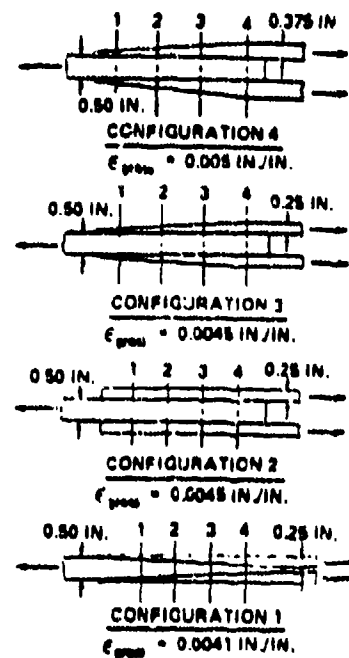
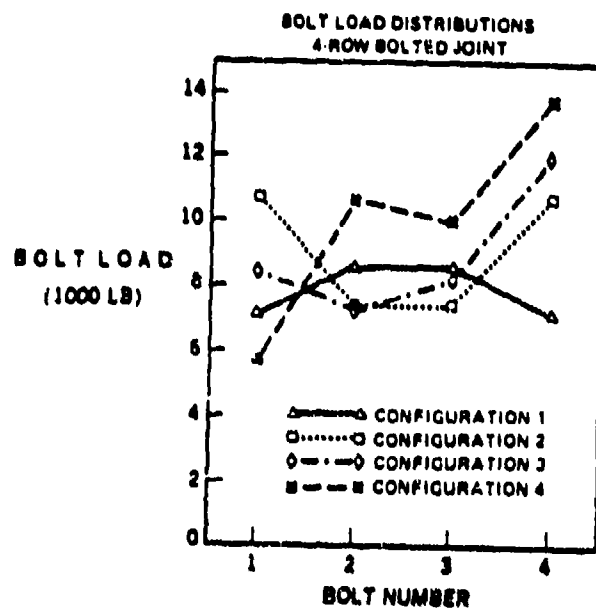


Figure 12. Effect of Joint Configuration on Fastener Load Distribution (Reference 9).

assumed in bolted ductile metals. This adversely influences the failure load for bolted laminates, unless thickness tapering or other configuration changes are introduced.

(7)

2.3.3 Bearing and By-Pass Loads at an Isolated Fastener Location

Figure 13 illustrates the bearing and by-pass loads, and the interaction between them, at an isolated fastener location in a bolted laminate. The failure of the bolted plate is generally assumed to coincide with the failure at the most critical fastener location. The identification of the most critical fastener location requires a knowledge of the load distribution among the fasteners, and an understanding of the interaction between the bearing and by-pass loads at a fastener location (Figure 13).

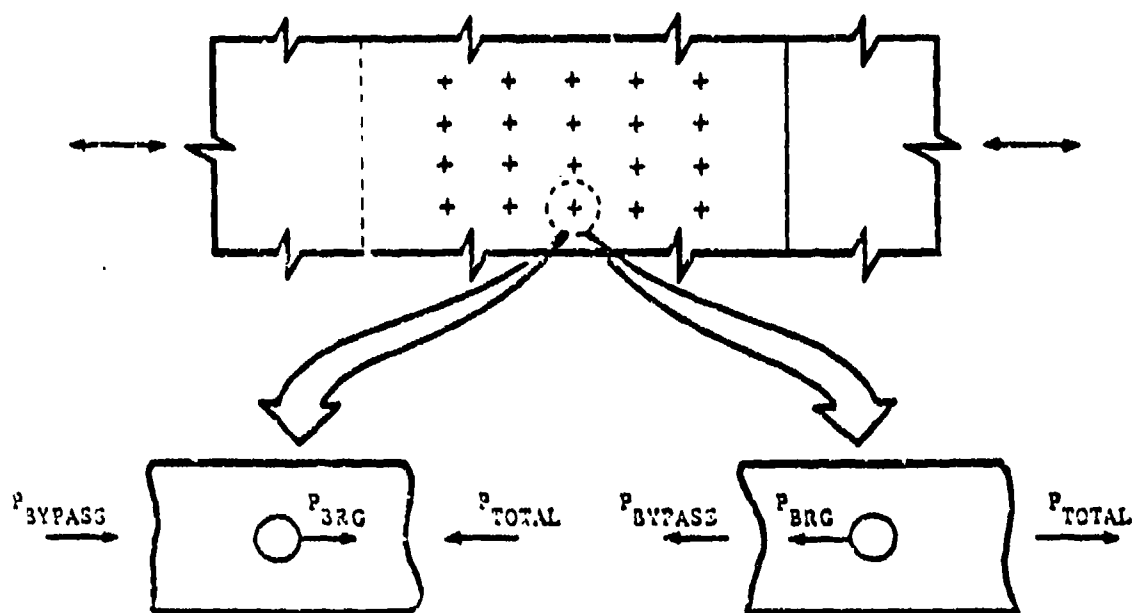
In ductile metals, minimal interaction is assumed between the bearing load and the by-pass load. However, in composites, a significant interaction has been demonstrated between the two loads under tensile loading (see Figure 13). Only a minimal interaction is observed under compression (see Figure 13). The open hole and bearing strengths of laminates (under tension and compression) are dependent on the laminate layup. The bearing stress at failure is also dependent on the edge distance (geometry) of the bolted laminate when its layup contains more than 40% of 0-degree plies.

Under tensile loading, an increase in the bearing stress reduces the by-pass stress value at failure in bolted laminates.

(8)

Under compressive loading, a minimal or negligible interaction between the bearing and by-pass loads is observed in bolted laminates

(9)



Bearing Stress at Failure

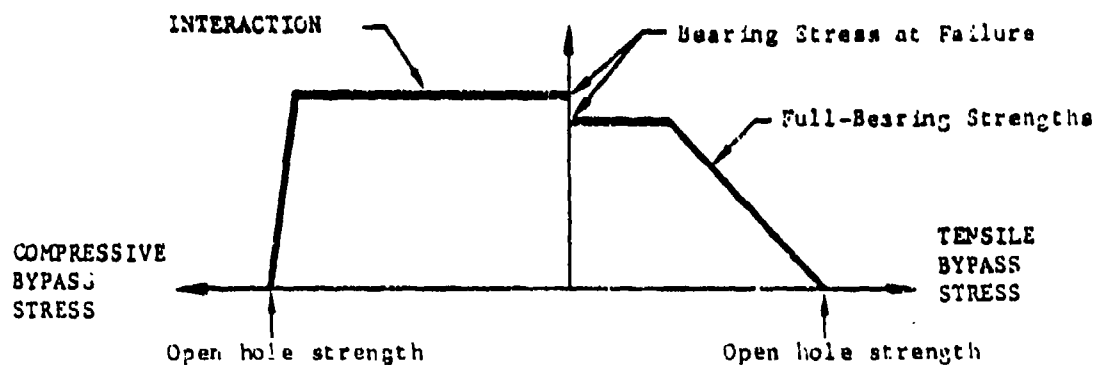
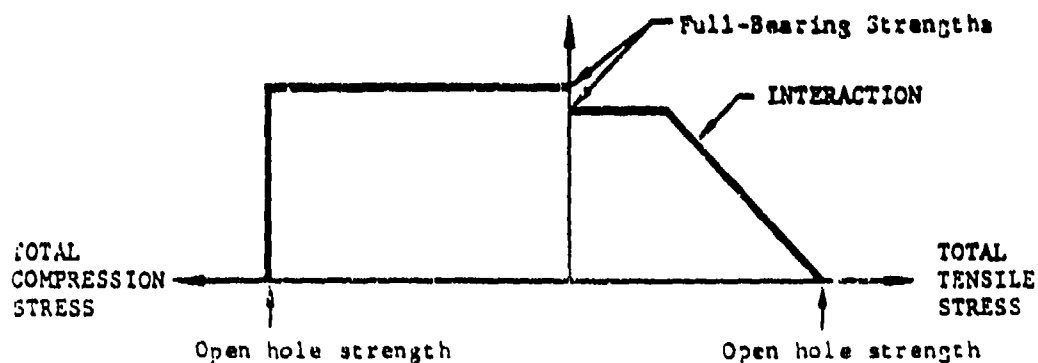


Figure 13. Interaction Between Bearing and by-Pass loads at a Fastener Location.

2.4

Failure Modes in Bolted Laminates

Bolted laminates exhibit one or more among a variety of failure modes, depending on their layup and geometry, the fastener type and the loading configuration. Figure 14 presents the basic failure modes observed in bolted laminates and possible fastener or fastener-induced failures. In the design of bolted laminates using the SAMCJ computer code, only the net section, shear-out and bearing modes of failures in the laminate are considered, and fastener-related failures are assumed to be precluded a priori. Net section and shear-out failures lead to catastrophic joint failures, while bearing failure is generally non-catastrophic. Critical, highly-loaded structural joints should, therefore, be designed to fail in a bearing mode.

Ensuring that fastener-related failures are predicted, highly-loaded structural joints must be designed to fail in a bearing mode to avoid the catastrophic failures induced by net section and shear-out modes of failure.

(10)

2.5

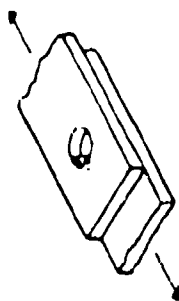
Fastener Type, Material and Installation Variables

In selecting fasteners for bolted composite structures, many variables have to be considered. These are briefly discussed below.

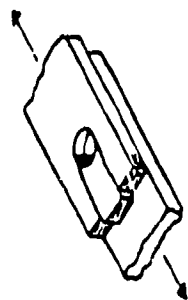
2.5.1

Fastener Type

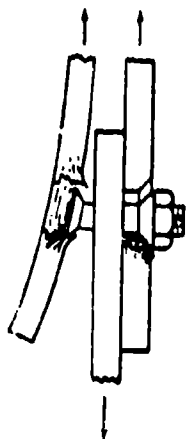
Fasteners are available in different forms for different applications, and are broadly classified as protruding head fasteners or countersunk (flush head) fasteners. Countersunk fasteners generally have a 100 degree head angle, and are referred to as tension head or shear head fasteners based on the countersunk depth. Special fastener types include hi-lok, big foot, Jo-bolt, Eddie-bolt, k-Lobe, composite fasteners, etc. (Reference 11).



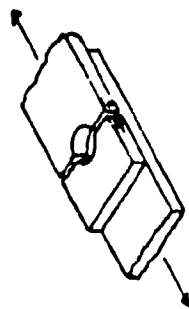
BEARING FAILURE



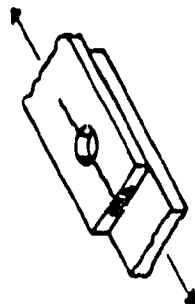
SHEAR-OFF FAILURE



BOLT PULLING THROUGH LAMINATE



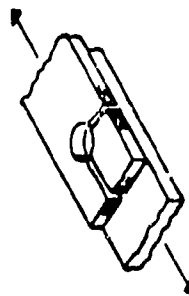
NET TENSION FAILURE



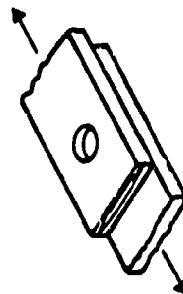
CLEAVAGE FAILURE



BOLT FAILURE



COMBINATION FAILURE



DELAMINATION FAILURE

Figure 14. Basic Failure Modes in Bolted Laminates and Fastener-Related Failures.

The joint location influences the selected fastener type and introduces sealing requirements (see Section 2.1). The three guidelines corresponding to this are repeated below:

Flush head (countersunk) fasteners should be used on aerodynamic surfaces to maintain contour smoothness. (1)

In fuel containment areas, the fastener locations must be sealed to be leak-proof and to prevent arcing in the fuel cell in the event of a lightning strike. (2)

In areas of restricted accessibility, blind fasteners must be used. (3)

Tension head countersunk fasteners have a larger countersunk depth than shear head countersunk fasteners. Tension head fasteners, therefore, rest over a larger area of the bolted plate, and carry the load primarily in tension along the fastener axis. Shear head fasteners have a smaller countersunk depth, and carry the load primarily in shear over the fastener cross-section. Consequently, tension head fasteners are capable of carrying larger loads than shear head fasteners. But, when the countersunk depth exceeds approximately 70% of the bolted plate thickness, the fastener effectiveness is reduced due to the local "knife edge" effect, influencing the selection of the fastener type.

Tension head fasteners are preferred over shear head fasteners when the countersunk depth is below approximately 70% of the bolted plate thickness. (11)

2.5.2 Fastener Material

The main considerations in the selection of the fastener material are its compatibility with the bolted plate material and its mechanical properties. Galvanic corrosion is a problem when steel or aluminum is used adjacent to graphite/epoxy composites, especially in a salt spray atmosphere (see Table 1, Figure 15 and Reference 12). Titanium does not corrode when it is in contact with graphite/epoxy composites. The compatibility of other materials with graphite/epoxy composites is rated in Table 1. Consequently, titanium fasteners are preferred for use in bolted composite structures. Also, a corrosion barrier is generally introduced between bolted composite and metallic parts, if the metal is steel or aluminum (see Figure 15).

Titanium fasteners are preferred for use with graphite-reinforced composites. Steel and aluminum fasteners are not recommended for use with these composites due to their corrosion susceptibility.

(12)

2.5.3 Fastener Size

The fastener size is generally selected to preclude excessive fastener bending effects that could reduce its load transfer capability and induce premature fastener failure. As a general rule, the ratio of the fastener diameter (D) to the bolted plate thickness (t) should be greater than 1 (see Figure 16).

The fastener diameter must be larger than the thickness of either bolted plate.

(13)

2.5.4 Fastener Fit and Hole Quality

Structural parts that are mechanically fastened together are drilled in accordance with established process specifications. Nevertheless, the presence of flaws at fastener locations is commonplace. These flaws include improper fastener seating,

TABLE 1. GALVANIC COMPATIBILITY OF FASTENER MATERIALS WITH COMPOSITES (REFERENCE 12).

Fastener Material	Compatibility with Graphite/Epoxy Composites
Titanium and its alloys	Very Good
MP-35N, INCO 600 (Nickel, Cobalt alloys)	Good
A286, PH13-8MO (Molybdenum alloys)	Acceptable
Monel	Marginal
Low Alloy Steel	Not Compatible
Silver Plate, Chrom. Plate	Adequate with/A286, PH13 13-8MO
Cadmium or Zinc Plate	Not Compatible
Aluminum or Magnesium Alloys	Not Compatible

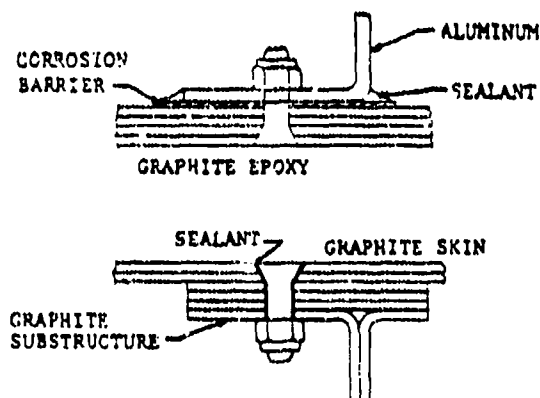


Figure 15. Galvanic Compatibility and Corrosion Prevention.

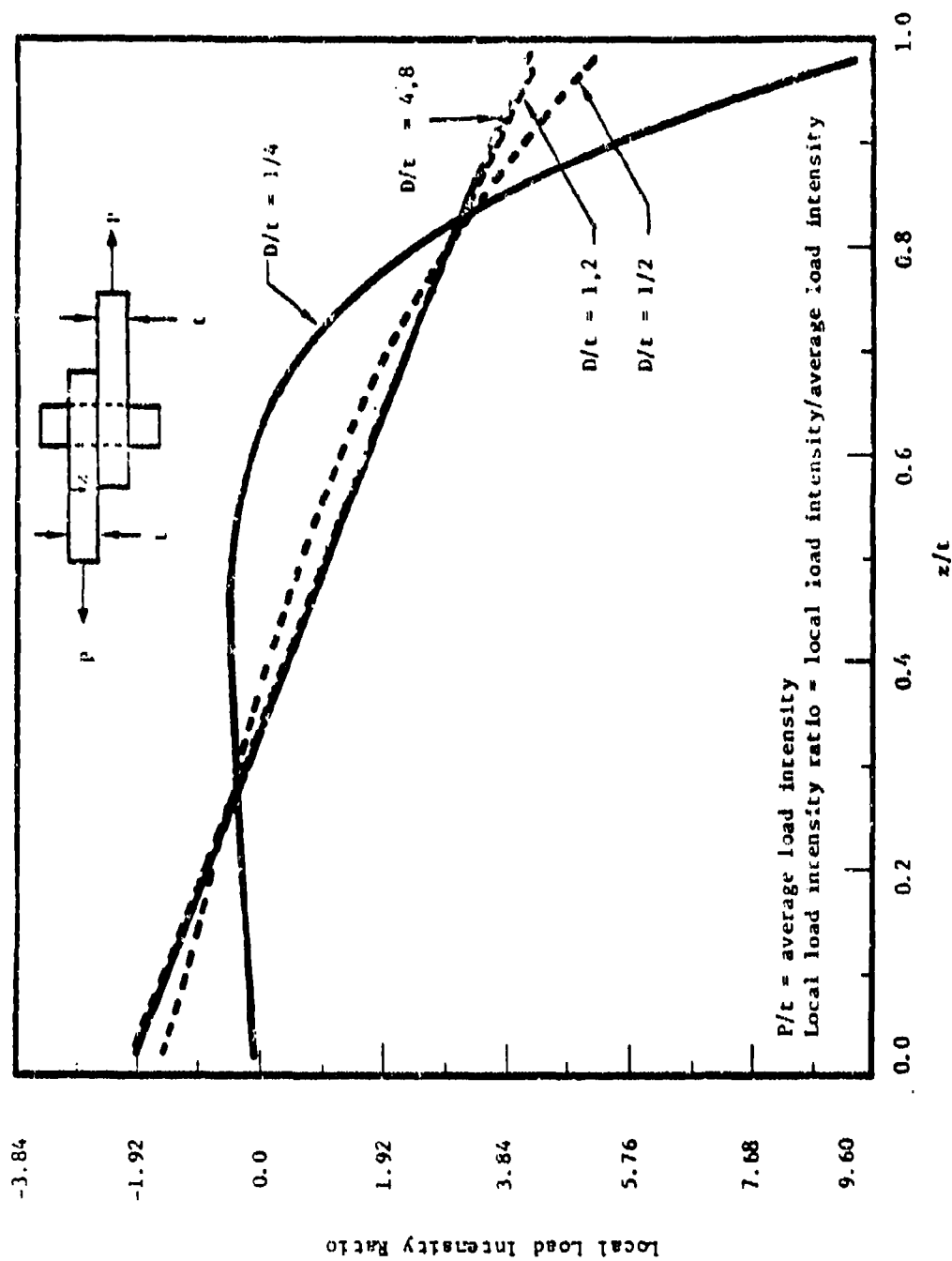


Figure 16. Effect of Fastener Size on the Load Distribution (Reference 6).

cratering of the hole boundary, broken and separated fibers at the drill exit side, delaminations near the exit surface, and a slight tilt (<10 degrees) in the hole axis away from the normal to the bolted plate (Reference 13). Interference fit of fasteners will also affect hole quality and influence the efficiency of the joint. The effects of interference fits and fastener hole flaws were studied in Reference 13 (see Table 2). A summary of the results is presented below:

Interference fastener fits (up to 0.008 inch of interference) induce negligible tensile strength losses. Nevertheless, they are generally not recommended due to installation problems and their effect on hole quality.

(14)

If the countersunk fastener seating (assuming 50% of the bolted plate thickness to be the nominal countersunk depth) is increased beyond 80% of the bolted plate thickness, the joint strength is decreased considerably (20 to 50%).

(15)

If the countersunk hole axis is at least 10 degrees away from the normal to the bolted plate, significant joint strength losses result (over 20% for a 10 degree tilt).

(16)

Other flaws (exit side broken fibers and delaminations, less than a moderate level of porosity in bolted laminates, holes offset by less than 0.005 inch, etc.) at fastener locations introduce negligible joint strength losses ($<10\%$).

(17)

2.5.5

Fastener Torque-Up

Static and fatigue tests on composite-to-metal joints

TABLE 2. EFFECTS OF FLAWS AT FASTENER HOLE LOCATIONS (REFERENCE 13).

	Percent Change in Strength*			
	COMPRESSION		TENS (OH)	
	RTW	250W	RTD	250W
OUT-OF-ROUND HOLES				
• 50/40/10 LAMINATE	-	-	<2.0	-
• 30/60/10 LAMINATE	-	-	-4.8	-
BROKEN FIBERS EXIT SIDE OF HOLE				
• SEVERE	-3.5	-12.2	-9.5	-
• MODERATE	-6.2	<2.0	-4.9	-
POROSITY AROUND HOLE				
• SEVERE	-12.1	-32.8	<2.0	-
• SEVERE WITH FREEZE-THAW	-13.3	-	-	-
• MODERATE	-5.4	-17.9	-	-
• MODERATE WITH FREEZE THAW	-7.9	-	-	-
IMPROPER FASTENER SEATING DEPTH (50% OF NOMINAL)				
• 80% THICKNESS	-	-	-23.2	-
• 100% THICKNESS	-	-	-56.9	-
TILTED COUNTERSINKS				
• AWAY FROM BEARING SURFACE	-	-20.0	<2.0	-
• TOWARD BEARING SURFACE	-	-22.7	-23.7	-
INTERFERENCE FIT TOLERANCES (INCH)				
• 50/40/10 @ 0.003	-	-	<2.0	+14.7
• @ 0.008	-	-	<2.0	+11.2
• 30/60/10 @ 0.003	-	-	<2.0	+2.4
• @ 0.008	-	-	<2.0	<2.0
FASTENER REMOVAL AND REINSTALLATION				
• 100 CYCLES	-	-7.4	<2.0	-

* RTD - Room Temperature, Dry; RTW - Room Temperature, Wet; 250W - 250°F, Wet;
Wet - 0.86% Moisture by Weight.

were conducted in Reference 13, varying the fastener torque-up value from 0 in-lb to 160 in-lbs. Fastener torque-up significantly improved the static strength of the joint (15 to 30%), and its fatigue life at a selected stress level. Similar results were observed in Reference 14. Under fatigue loading, the torque-up inhibits the initial growth of local failures in the joint, and the results in a more abrupt fatigue failure due to excessive hole elongation than a joint with no applied torque.

Fastener torque-up increases the static strength of a joint and its fatigue life at a selected stress level.

(18)

2.6 Bolted Laminate Properties

The basic material and its layup (stacking sequence) in bolted laminates influence the joint performance considerably. When graphite/epoxy laminates are bolted to metallic substructures, galvanic corrosion must be addressed (see Figure 15 and Table 1). For example, a corrosion barrier like a glass/epoxy layer must be used between graphite-reinforced composites and aluminum substructures.

When graphite-reinforced composites are bolted to metallic substructures, corrosion barriers must be introduced if the metal is not compatible with the composite material (see Table 1).

(19)

The bolted laminate layup is generally denoted by the percentages of plies with fiber orientations of 0, + or -45 and 90 degrees, with respect to the primary loading direction, for most structural laminates. The envelope within which a bearing failure mode and the maximum bearing strength are realized is shown in Figure 17. Within this envelope, the strength is independent of the actual stacking sequence. This assumes a laminate width-to-fastener diameter ratio (W/D) of at least 4, and an edge distance (E) of at

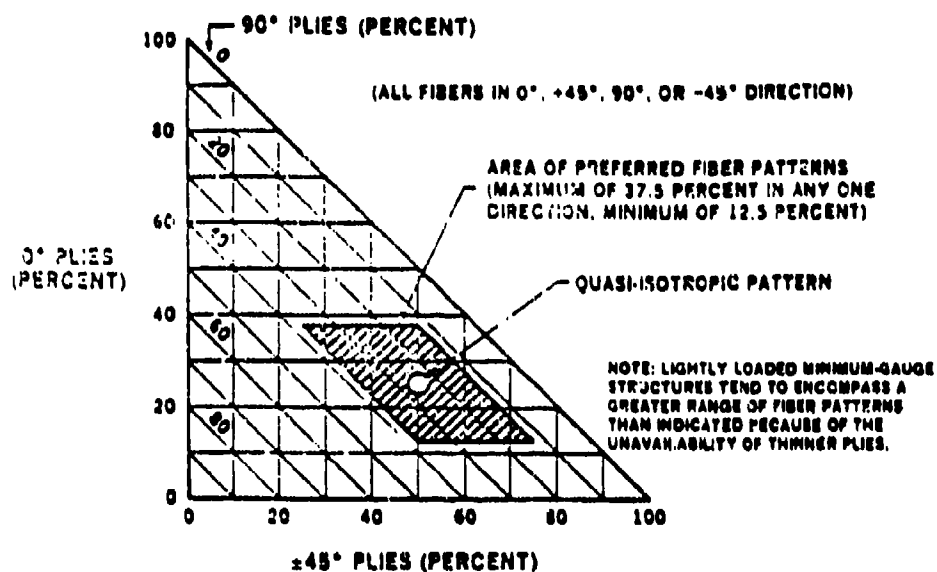


Figure 17. The Envelop of Bolted Laminate Layups for Realizing a Bearing Mode of Failure and the Maximum Bearing Strength (Reference 10).

least 3D. When the percentage of 0 degree plies exceeds 40, a shear-out mode of failure is introduced, reducing the bearing stress value at failure. Section 2.7 presents the effects of fastener spacing and the geometry of a bolted plate on its strength.

The bearing strength of a laminate is maximum when its layup contains less than 10% each of 0, + or -45 and 90 degree plies. The corresponding failure occurs in a bearing mode.

(20)

In addition, the individual plies must be arranged such that adjacent plies have different fiber orientations. If the stacking sequence contains groups of plies with identical fiber orientations, delamination-related failures will occur and reduce the joint strength.

Ply with different fiber orientations should be interspersed within the laminate, to the maximum possible extent, to minimize delamination-induced strength losses. Group of identical plies should not exceed 0.02 inch in thickness.

(21)

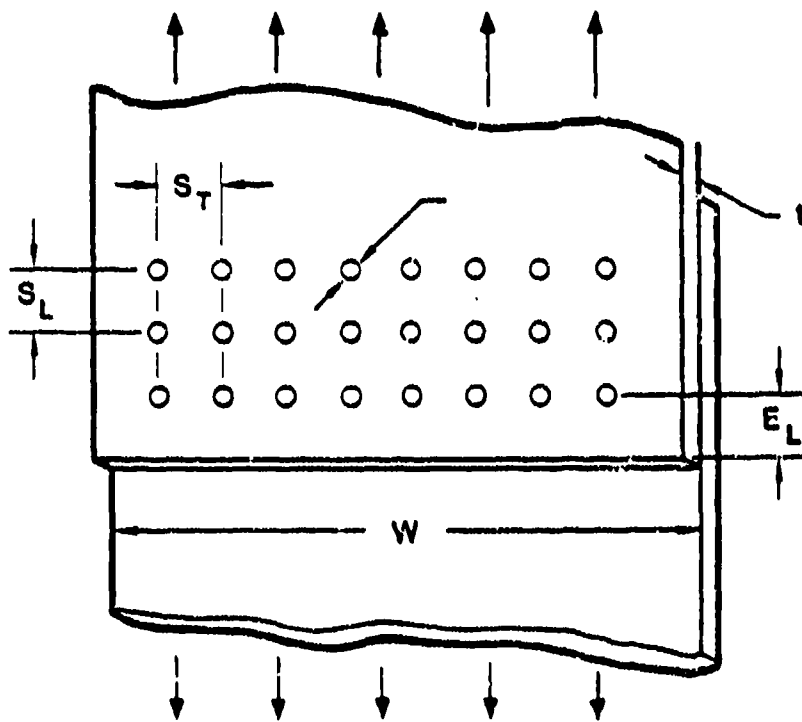
2.7 Fastener Spacing and Arrangement

The geometrical parameters that define the fastener spacing and the fastener arrangement in a bolted plate are illustrated in Figure 18. E is the edge distance, S_L and S_T are the fastener spacings in the loading and transverse directions, and $W = S_T$ for a single fastener joint. The effects of these geometrical parameters were studied in References 8, 13 and 14. The results are summarized below:

The bearing and net section strengths decrease when the fastener size increases (see Figure 19).

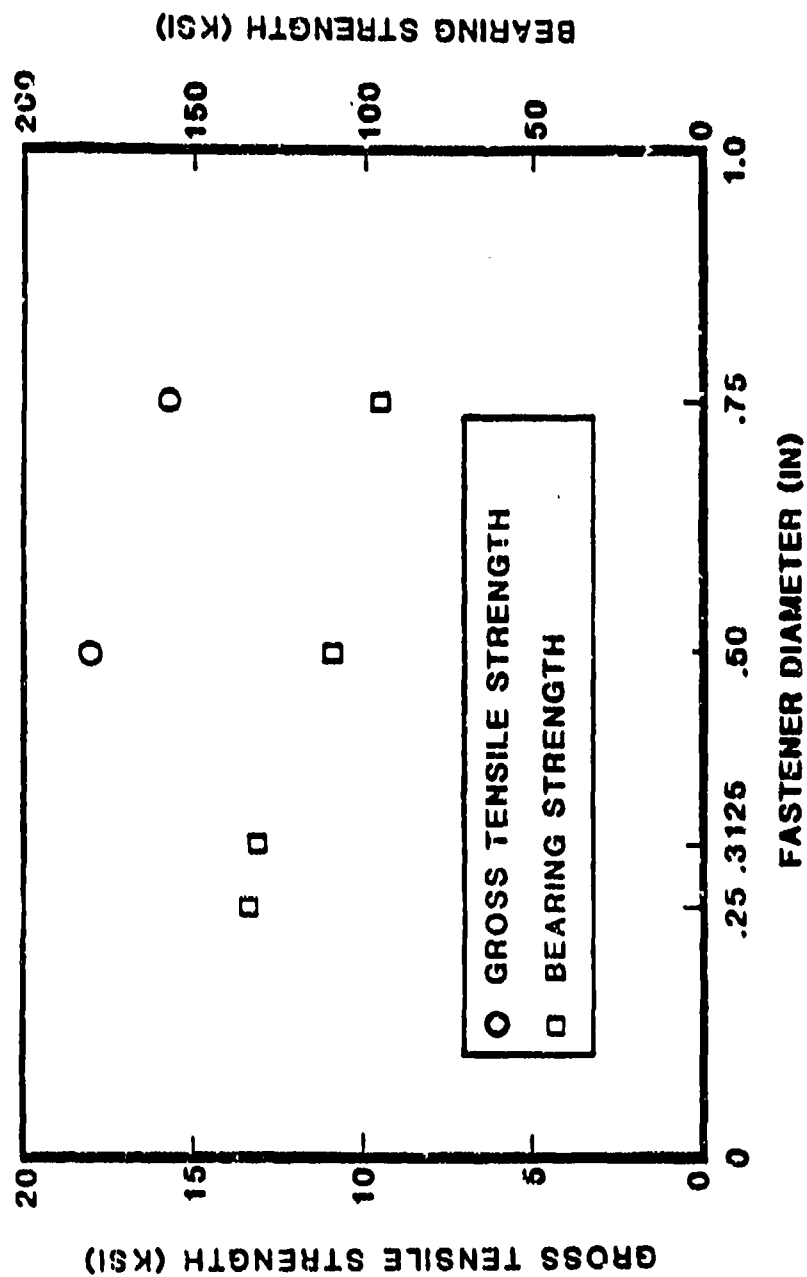
(22)

The bearing stress at failure decreases



PREDOMINANT LOAD DIRECTION

Figure 18. Geometrical Parameters for a Bolted Plate.



NOTE: 20-Ply, 50/40/10, AS1/3501-6 Layup; 0.31 in. Aluminum Plate;
Torque = 100 in-lbs, E/D = 3, W/D = 6, Protruding Head Steel
Fasteners; RTD; Net Section Strength = 1.2 x Gross Strength

Figure 19. Effect of Fastener Size on the Tensile Response of Composite-to-Metal Joints in Single Shear.

significantly when E/D is reduced below 3 (see Figure 20). A bearing mode of failure is observed only when $E/D > 4$, and the percentage of 0 degree plies is less than 40. A shear-out mode of failure results when $E/D < 3$, or when the percentage of 0 degree plies is > 40 .

(23)

The bearing stress at failure decreases significantly when S/D (W/D for a single-fastener joint) is reduced below 4 (see Figure 21). When $E/D > 3$, $W/D > 4$, and the percentage of 0 degree plies is below 40, a bearing mode of failure occurs. When $W/D < 4$, a net section failure occurs in the same laminate.

(24)

When the fastener spacing in the loading direction (S_L/D) is decreased below 4, the joint strength decreases due to stress concentration interaction (see Figure 22). The same effect is observed with S_T/D (see Figure 21).

(25)

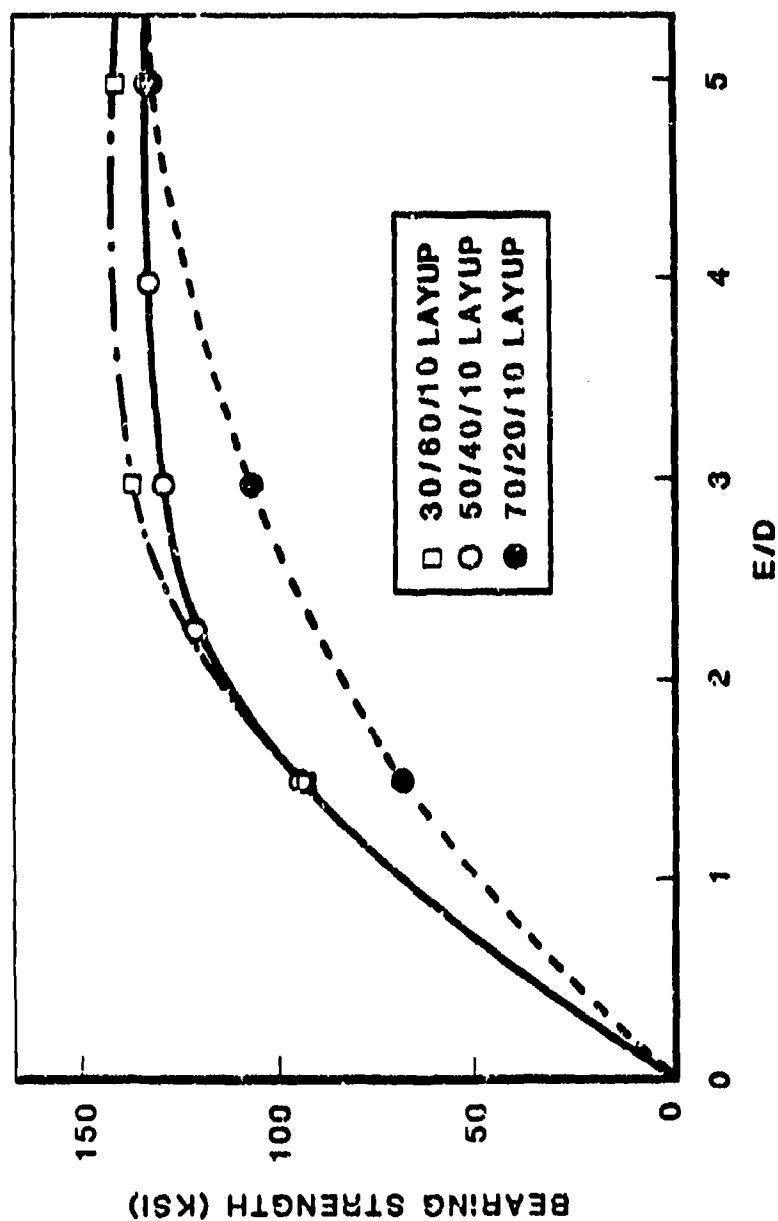
In summary, ensure that $D/t > 1$, $E/D > 3$, W/D (S_T/D) > 4 , $S_L/D > 4$, and the percentage of plies in any orientation is < 40 , to achieve a bearing failure mode and to realize the maximum joint strength.

(26)

2.8

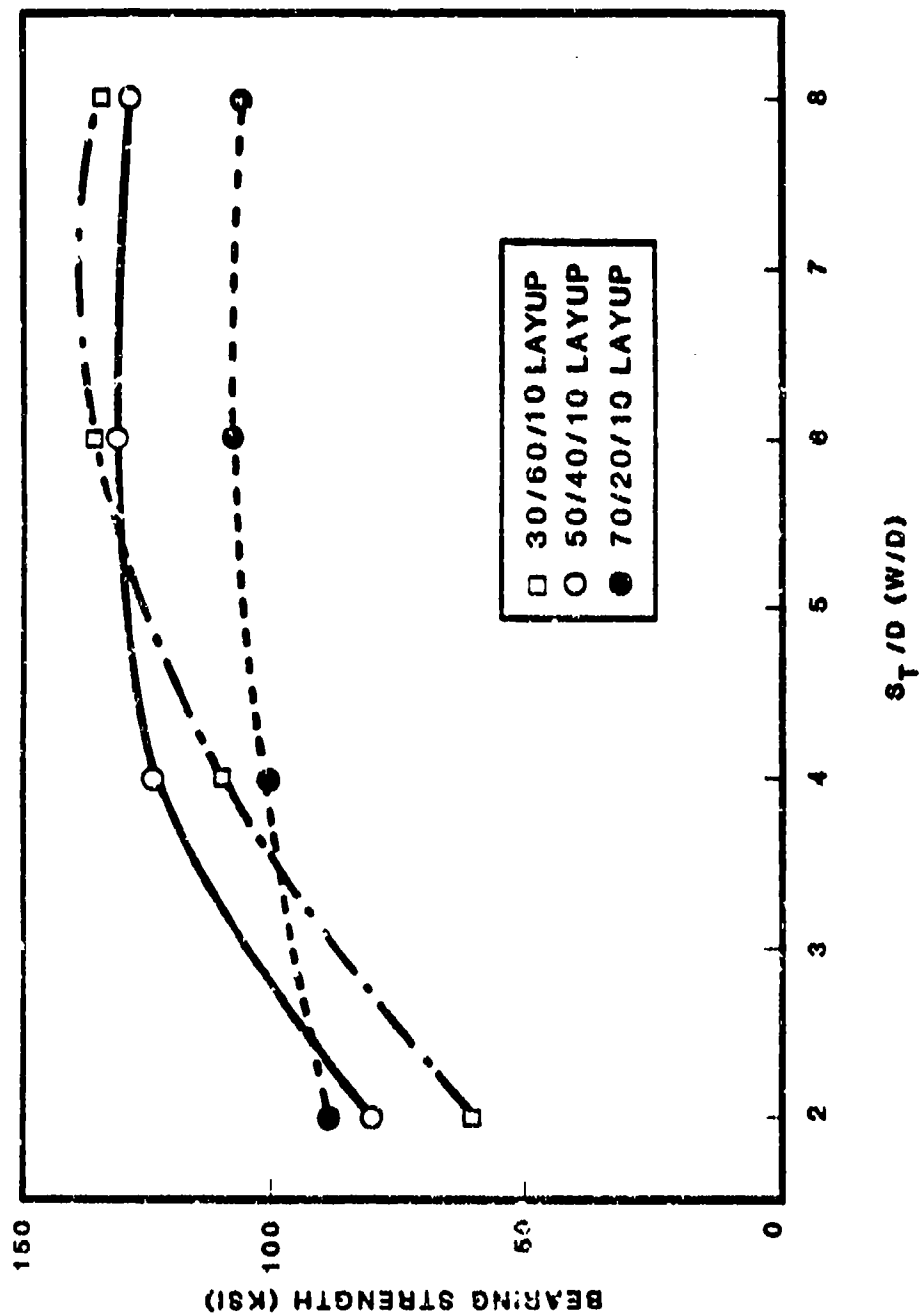
Joint Tailoring for Maximum Efficiency

The design of a joint should achieve the following objectives to be considered efficient: (1) It should be capable of transferring the design ultimate loads without failing any member; (2) It should possess the design life when subjected to the design spectrum fatigue loading; (3) It should be the least weight design that meets (1) and (2); and (4) The complexity of the design concept



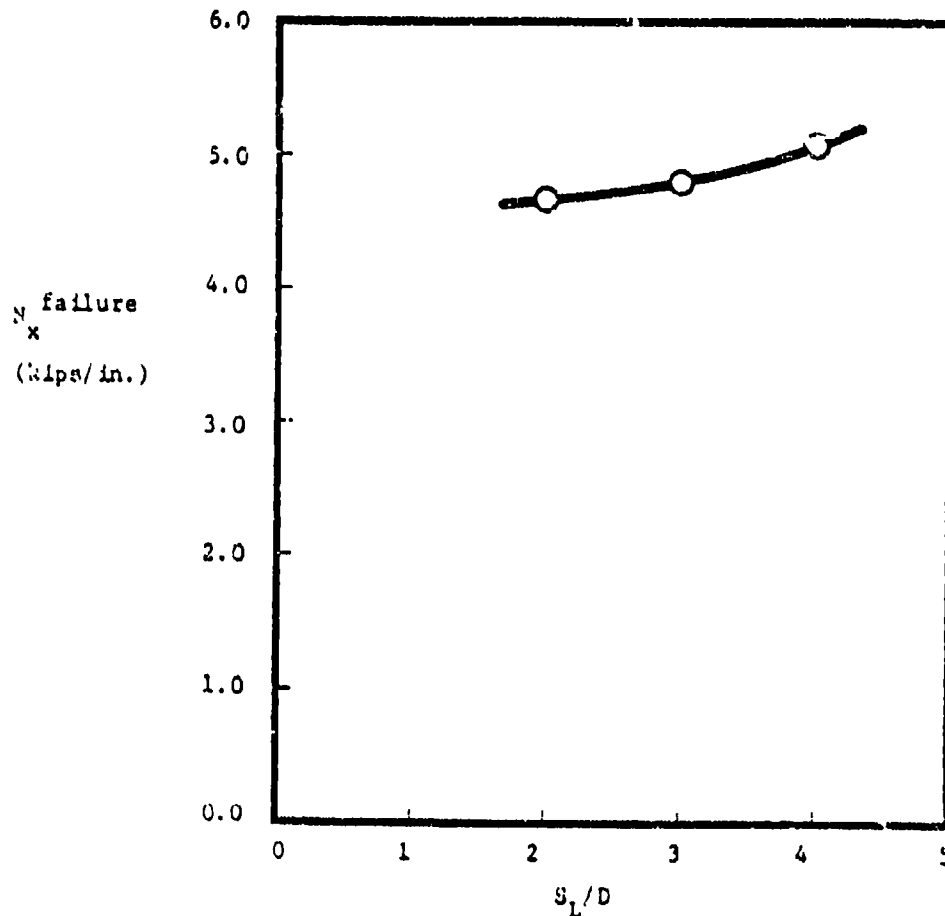
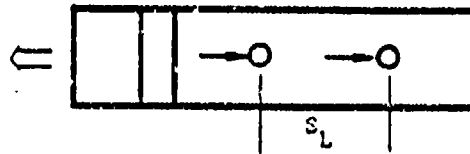
NOTE: Composite-to-metal joints in single shear; 20-ply AS1/3501-6 layups; 0.31 in. aluminum plate; $W/D = 6$; Torque = 100 in-lbs; Protruding Head, steel fastener; $D = 5/16$ in.

Figure 20. Effect of E/D on the Bearing Strength of Bolted Laminates.



NOTE: Composite-to-metal joints in single shear; 20-ply, AS13601-6 layups;
 0.31 in. aluminum plate; $E/D = 3$; Protruding head steel fastener;
 $D = 5/16$ in.; Torque = 100 in-lb

Figure 21. Effect of W/D on the Bearing Strength of Bolted Laminates.



Note: Composite-to-metal, two fasteners-in-a-row joint; 20-ply, 50/40/10 layup; AS1/3501-6 graphite/epoxy; 0.31 in. aluminum plate, single shear; RTD; static tension; $S_L/D=W/L=6$; protruding head steel fastener; $D=5/16$ in.; $T=100$ in-lbs.

Figure 22. Effect of S_L/D on the Strength of Bolted Laminates.

should be controlled to aid producibility and maintainability of the structural joint.

A joint can be tailored to improve its efficiency. For example, when the number of fastener rows (a row being perpendicular to the primary loading direction) is increased, the peak load fraction is generally carried by the innermost or outermost fastener row (see Figures 11 and 12). If the failure mode at the critical fastener location is bearing or net section, the thickness and width of the bolted plate at that location will influence the joint failure load. In an efficient design, the width and the thickness of the bolted plates will be tailored such that every fastener location is equally critical (see Figure 5). The peak bearing stress at the design ultimate load level will be lowered to a level that ensures a minimal bearing/by-pass interaction, if possible (see Figure 13).

Some experimental concepts have also been demonstrated to be efficient joint tailoring concepts, despite the difficulty they introduce in applying the concept at the production level. An example is shown in Figure 23, where the 0 degree plies in the bolted skin are replaced by + and -45 degree plies in the joint region (Reference 15). This causes a smaller fraction of the running load to be transferred at the joint location, and also increases the local bearing strength. An alternative, equivalent concept would be to replace the stiffer material by a tougher material at the joint location. For example, graphite/epoxy plies can be replaced by aramid fiber/epoxy plies at the joint location. It is reiterated, though, that these validated tailoring concepts are difficult to implement in a production environment.

The geometry of bolted laminates must be tailored, in the width and thickness directions, to render every fastener location equally critical.

(27)

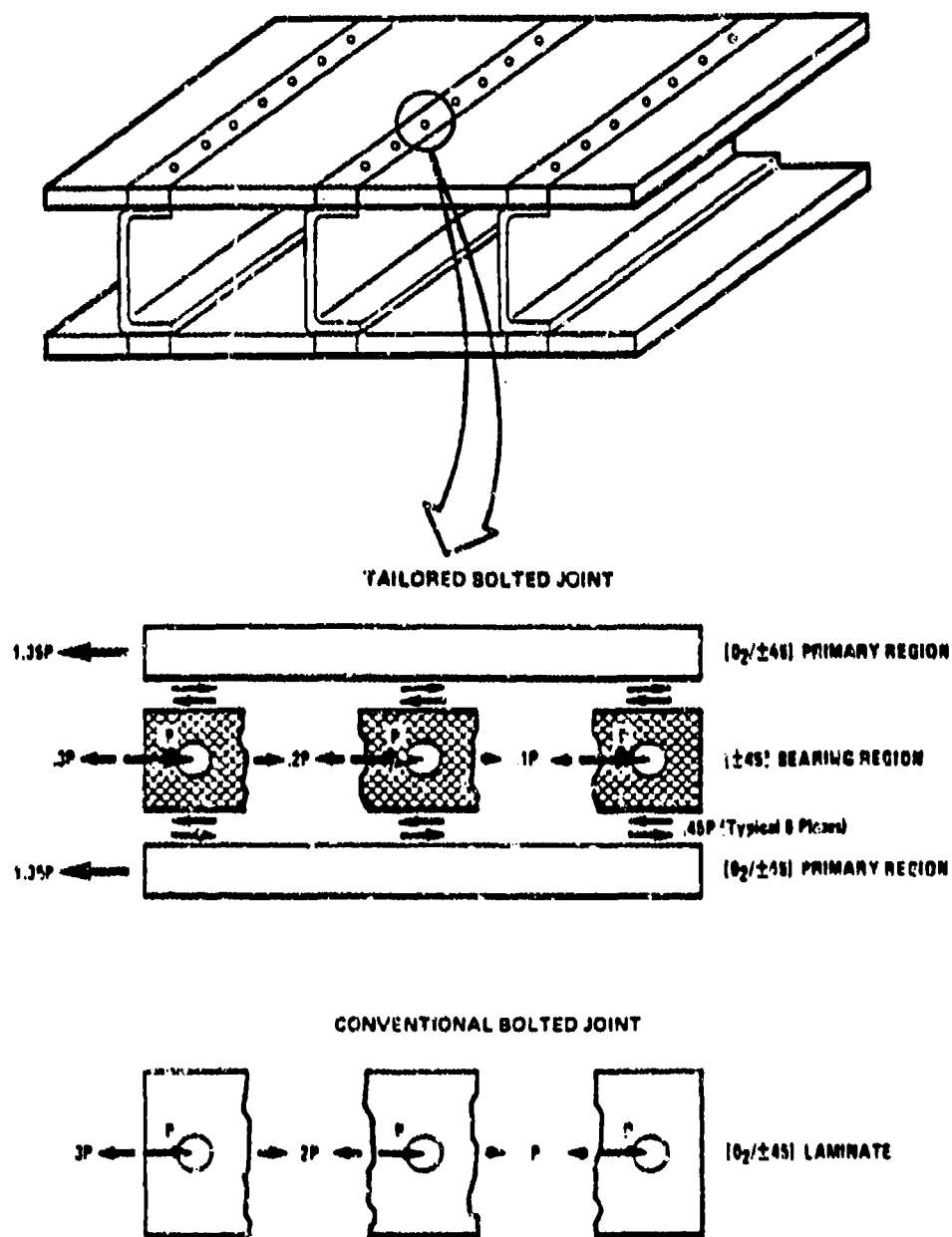


Figure 23. A Sample Tailored Joint.

The design of a bolted joint is currently based on an assumed design ultimate load level and a static strength analysis (see Section 3). The assumed design ultimate load level should account for durability considerations also. Generally, irrespective of the static failure mode, a bolted joint suffers fatigue failure via excessive hole elongation (bearing). This possible change in the failure mode from the static loading case to the fatigue loading case has been observed by many in the literature (see References 13 and 14).

If the joint statically fails in a bearing mode, it could suffer premature excessive hole elongation (fatigue failure) when subjected to the spectrum fatigue loading. Figures 24 and 25 present sample constant amplitude fatigue test results from Reference 14 for a fully reversed loading case ($R=-1$). Similar results should be used to approximately and conservatively estimate the fatigue life of a joint using a fatigue analysis (Miner's rule, for example). Based on the fatigue analysis, the bearing stress at the critical fastener location should be designed to be sufficiently lower than the static bearing strength, to ensure the design life of the joint. The final joint design, therefore, will be capable of statically transferring the design ultimate load, with the peak bearing stress value ensuring the design fatigue life.

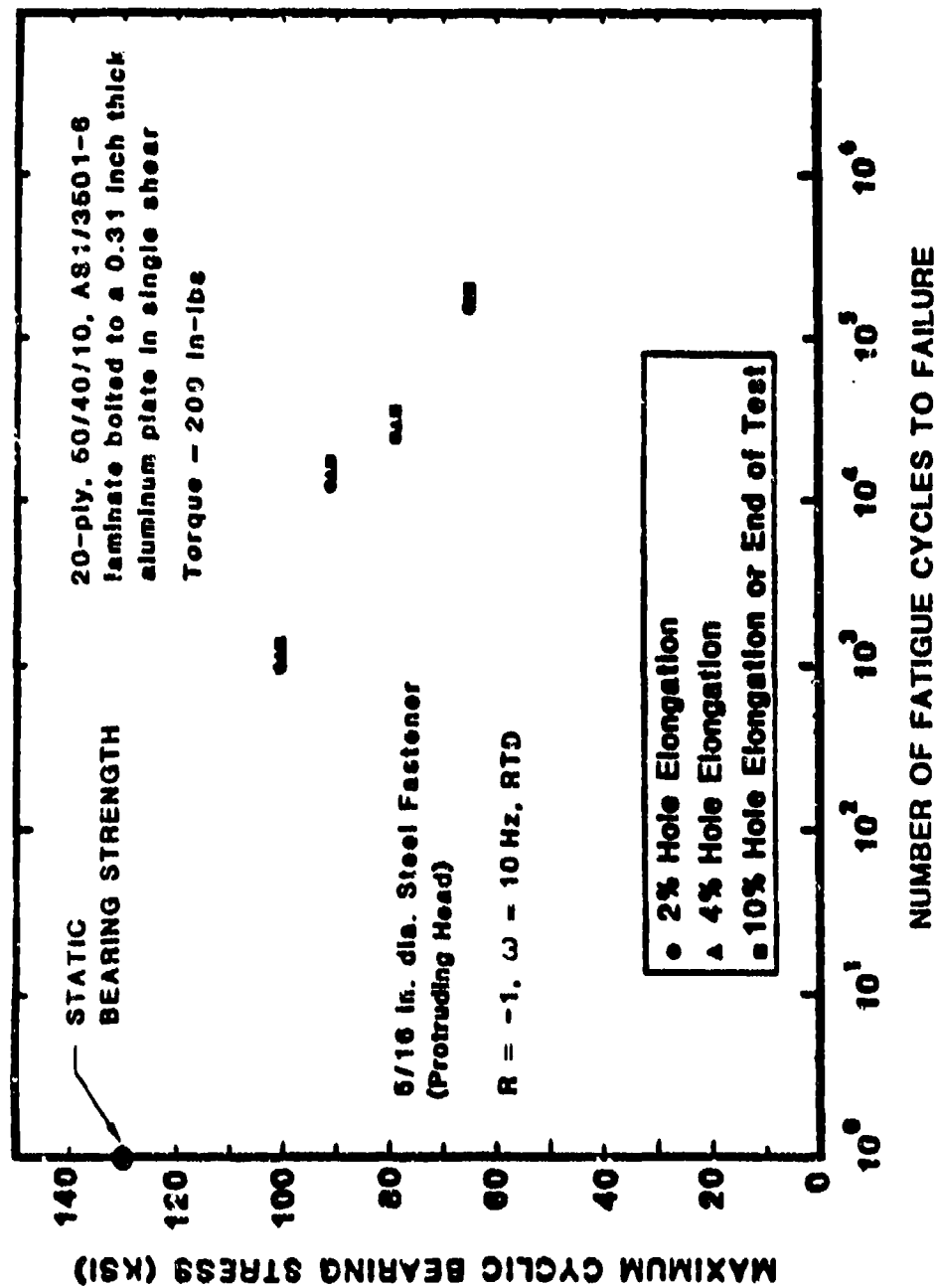


Figure 24. Effect of Maximum Cyclic Bearing Stress on the Number of R-1 Fatigue Cycles to Cause Specified Hole Elongations in a Bolted Laminate.

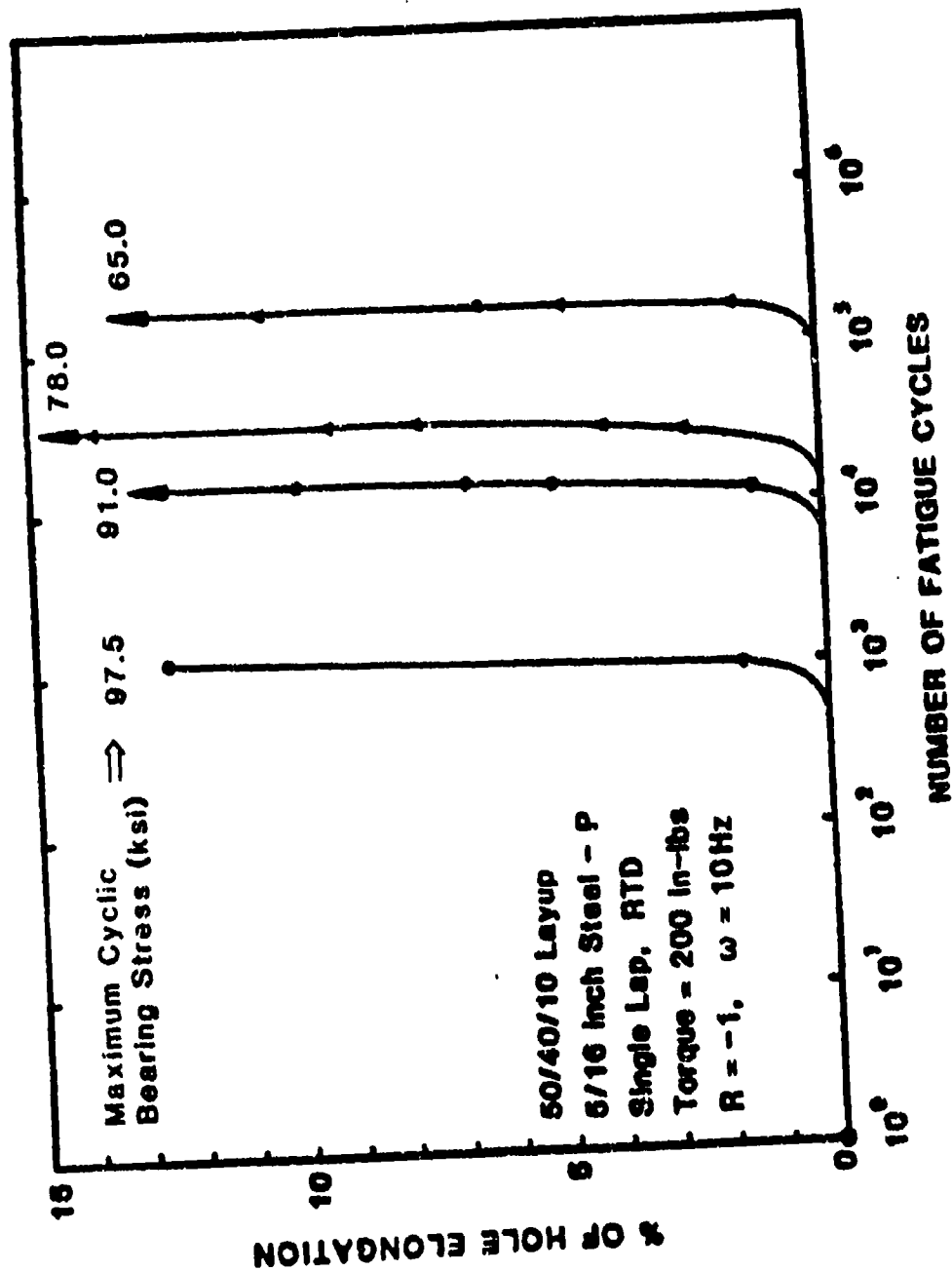


Figure 25. Effect of Maximum Cyclic Bearing Stress on the Hole Elongation Rate for a Bolted Laminate under $R=-1$ Loading.

SECTION 3

STRENGTH ANALYSIS OF BOLTED COMPOSITE STRUCTURES

As mentioned in Section 1.4, two computer codes were developed in this Northrop/AFWAL program to predict the strength of bolted joints containing a single fastener (SASCJ and SAMCJ) or multiple fasteners (SAMCJ). Most of the structural joints contain multiple fasteners, and SAMCJ is adequate for the design of these joints. SAMCJ is also capable of predicting the strength of single fastener joints, without accounting for the nonlinear joint load versus deflection behavior introduced by ply level failures. However, if the user wishes to interrogate an isolated fastener location, accounting for the nonlinear joint behavior due to progressive (two-stage) ply failures, the SASCJ code is useful. The reader is referred to References 6 and 7 for detailed descriptions of the SASCJ and SAMCJ analyses, respectively.

In the following sub-sections, brief descriptions of the analyses in the SASCJ and SAMCJ computer codes are presented, along with detailed instructions for the use of these analytical design tools.

3.1 Description of SASCJ Analysis

A two-dimensional anisotropic plate analysis that accounts for finite plate dimensions (FIGEOM), and a finite difference fastener analysis (FDFA), are incorporated into a progressive failure procedure to develop a strength analysis for single fastener joints in composite structures (SASCJ). An isolated fastener location in a bolted structures (see Figure 7) is primarily subjected to the loading shown in Figure 26. The general bolt bearing/by-pass situation can be analyzed as a superposition of an unloaded hole situation and a fully loaded hole situation (see Figure 26). The unloaded hole case is analyzed using the two-dimensional plate analysis (FIGEOM), and does not involve the

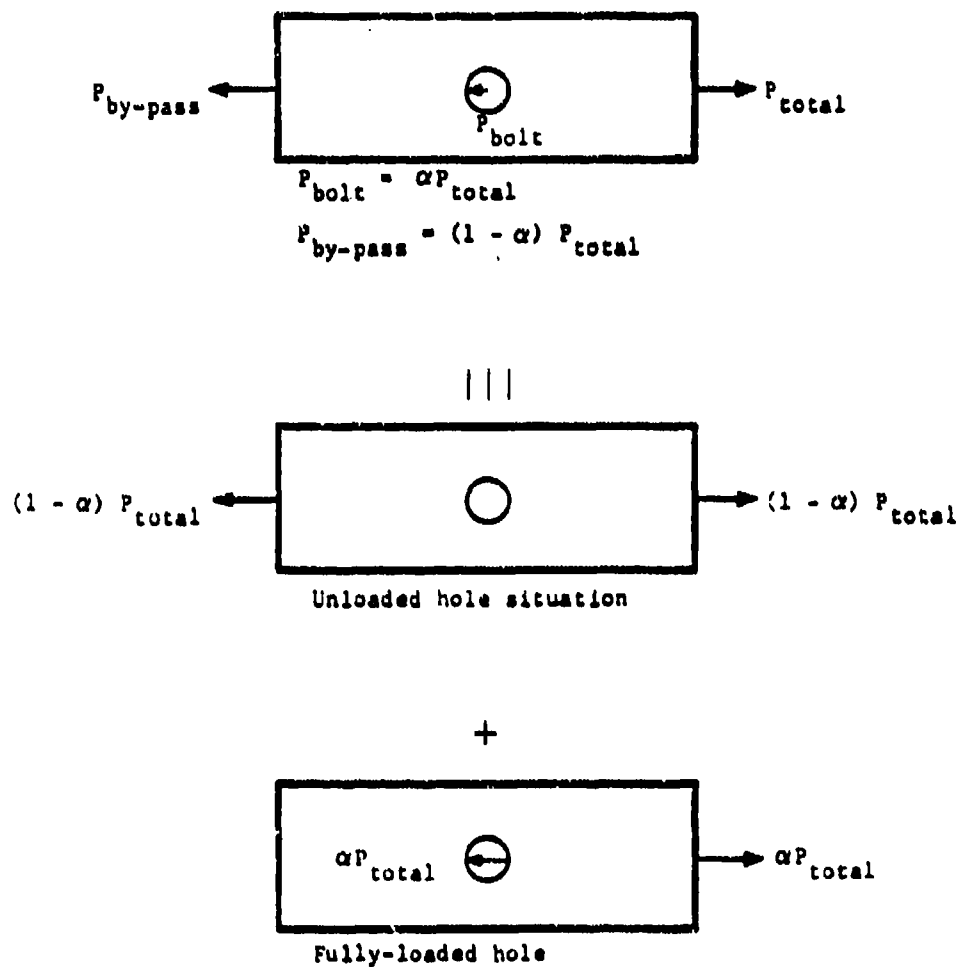


Figure 26. Schematic Representation of a General Single-Fastener Situation as a Superposition of Unloaded and Fully Loaded Hole Situations.

fastener analysis (FDFA). The fully-loaded hole situation is analyzed using a progressive failure procedure that predicts local ply failures and delaminations until the bolted plate cannot carry any additional applied load. The employed ply-failure criteria and the delamination criterion are discussed in Sections 3.1.3 and 3.1.4.

3.1.1 Strength Analysis Procedure for Fully-Loaded Holes

The strength of laminates with fully loaded holes is predicted using the procedure outlined in Figure 27. A two-dimensional stress analysis (FIGEOM), accounting for finite dimensions of the bolted plates, is initially performed on each bolted plate. Computed plate stresses are used to calculate the effective moduli of the various ply types in each bolted plate (see Reference 6). The inplane strains computed by the FIGEOM code are used to obtain the stress state in each ply. The ply stresses around the hole boundary are integrated to yield the bearing load in each ply (see Reference 6). The inplane stresses in each ply, per unit bearing load, are incorporated into selected failure criteria to compute the ply (bearing) loads corresponding to the various inplane failure modes.

The effective moduli and the ply bearing loads corresponding to the various failure modes, for all the plies in each bolted plate, are incorporated into the fastener analysis. The initial fastener analysis on the undamaged plates computes the distribution of the applied bearing load among the various plies. Comparing these ply loads with the stored failure values for inplane ply failures, the joint load corresponding to the earliest ply failure is obtained. The fastener analysis also computes approximate shear strain values at the interfacial locations between adjacent plies. Incorporating these into an interlaminar failure criterion, the joint load corresponding to the earliest interlaminar failure (delamination) is obtained. The smaller of the two joint loads, corresponding to the earliest inplane and interlaminar

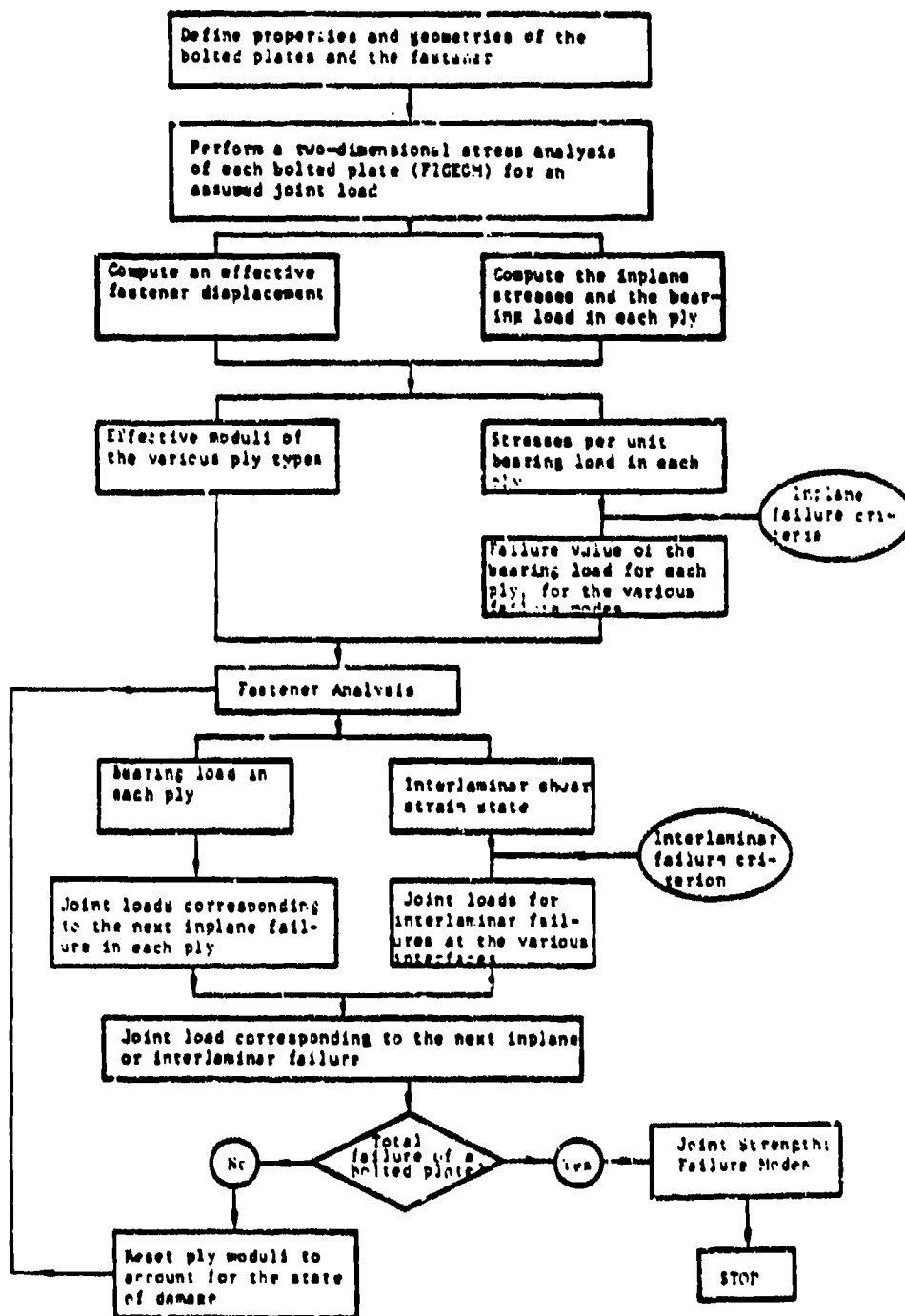


Figure 27. Flowchart for the Strength Analysis of Laminates with Fully Loaded Holes.

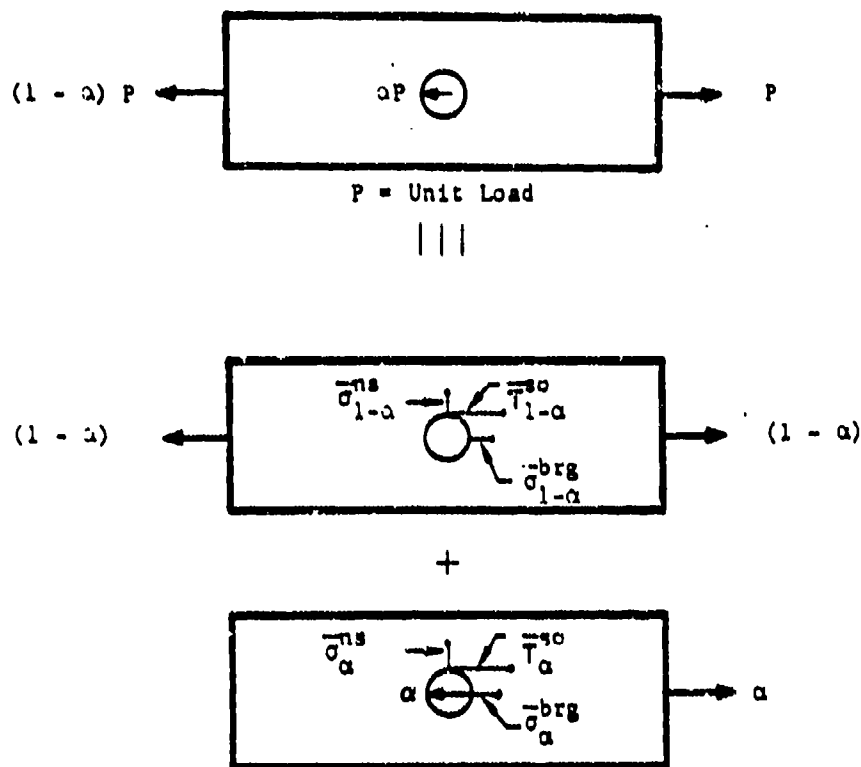
failures, determines the first failure in a bolted plate and the corresponding joint load value.

The effective moduli of the damaged plies are reset to appropriately represent the predicted failure modes. The revised moduli are incorporated into the fastener analysis, and the procedure is repeated to predict the next failure mode and the corresponding joint load. When any ply is predicted to fail totally, the analysis computes the redistribution of the corresponding joint load among the remaining effective plies, and determines if any other concomitant ply failure is precipitated. This process is repeated until one of the bolted plates becomes ineffective in transferring the applied load (joint failure).

The SASCF computer code is restricted to protruding head fasteners, and assumes that fastener failure is precluded. However, when a countersunk fastener is specified, SASCF assumes an appropriate boundary condition at the head location, and expects the user to input an equivalent (larger) uniform fastener diameter. It can analyze any combination of laminated and metallic plates, bolted together in a single-lap or double-lap configuration.

3.1.2 Strength Analysis Procedure for Partially-Loaded Holes

A general fastener location in a bolted plate transfers a fraction (α) of the total applied load via the fastener, the remainder ($1-\alpha$) being by-passed to the next fastener location (see Figures 7 and 26). In this case, the stress state at the fastener location is computed as a superposition of the stress states corresponding to the unloaded and fully-loaded hole situations. Figure 28, for example, presents a schematic representation of how the averaged stresses are obtained to predict net section, shear-out and bearing failures in the plies using average stress failure criteria. For a unit applied load, the averaged stresses in the laminate with an unloaded hole, when subjected to a load of $(1-\alpha)$, and the averaged stresses in the laminate with a fully loaded hole,



$$\bar{\sigma}^{ns} = \int_{D/2}^{D/2 + d_{ons}} \sigma_x(o, y) dy = \bar{\sigma}_{1-a}^{ns} + \bar{\sigma}_a^{ns}$$

$$\bar{\tau}^{so} = \int_0^{d_{oso}} \tau_{xy}(x, D/2) dx = \bar{\tau}_{1-a}^{so} + \bar{\tau}_a^{so}$$

$$\bar{\sigma}^{brg} = \int_{D/2}^{D/2 + d_{obrg}} \sigma_x(x, o) dx = \bar{\sigma}_{1-a}^{brg} + \bar{\sigma}_a^{brg}$$

Figure 28. Strength Analysis of Laminates with Partially-Loaded Holes using Average Stress Failure Criteria.

when subjected to a load of α , are computed separately and added. Incorporating the combined averaged stresses into the appropriate failure criteria, the applied load corresponding to a ply failure is computed.

In the case of fully loaded holes, progressive failure prediction involves the repetition of the fastener analysis with revised ply properties after every ply failure. The two-dimensional analysis (FIGEOM) is only carried out once. But, in the case of partially loaded holes, a ply failure will affect the unloaded and the fully loaded hole contributions to the local stresses. Hence, progressive failure prediction in the partially loaded case involves repeating FIGEOM and FDFA analyses after total ply failures.

3.1.3 Inplane Failure Criteria

The SASCJ code permits the user to select any of the following five failure criteria for the prediction of ply failures based on inplane stresses and strains: (1) point stress failure criterion, (2) average stress failure criterion, (3) maximum (fiber directional) strain criterion, (4) Hoffman criterion, and (5) Tsai-Hill criterion. The first two criteria predict three modes of failure in each ply-net section, shear-out and bearing. The maximum strain criterion predicts ply failure based on fiber failure. The Hoffman and Tsai-Hill criteria predict ply failure accounting for biaxial stress interaction that is ignored by the first three criteria.

The point stress failure criterion predicts net section, shear-out and bearing failures when the appropriate stress components at selected locations attain unnotched specimen failure values (see Figure 29). a_{ons} , a_{oso} and a_{obrg} are called characteristic distances. When $\sigma_x(0, D + a_{ons})$ exceeds the unnotched tensile or compressive strength of the ply, as appropriate, a net section ply failure is predicted. When $\sigma_x(D + a_{obrg}, 0)$ exceeds the unnotched compressive strength of the ply, a bearing mode of ply failure is

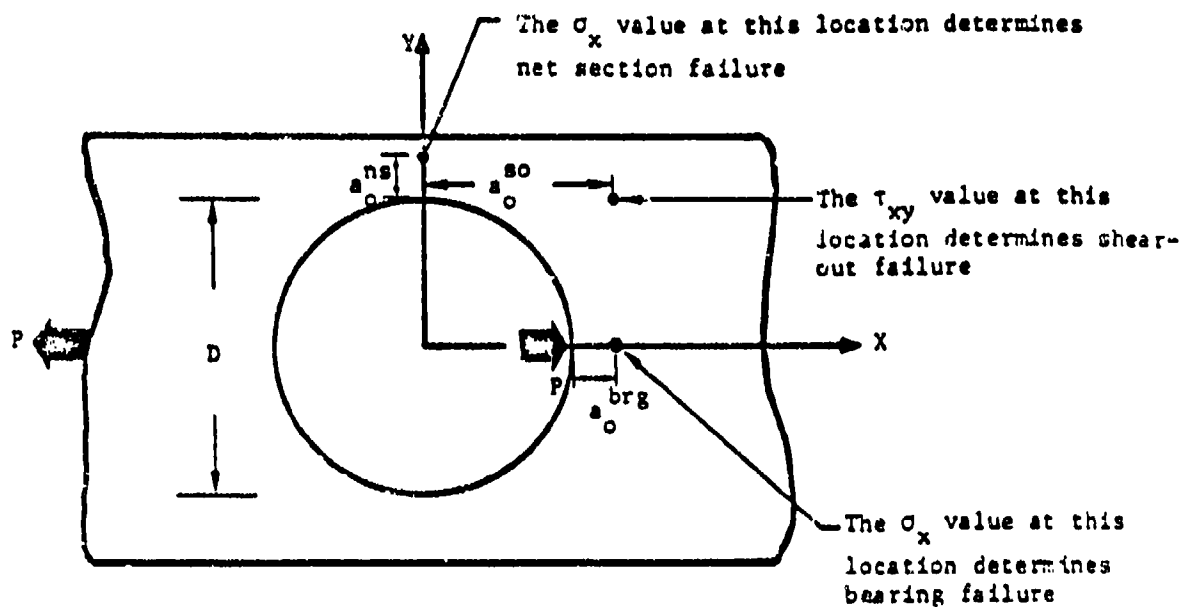


Figure 29. The Characteristic Distances used in the Point Stress Failure Criteria

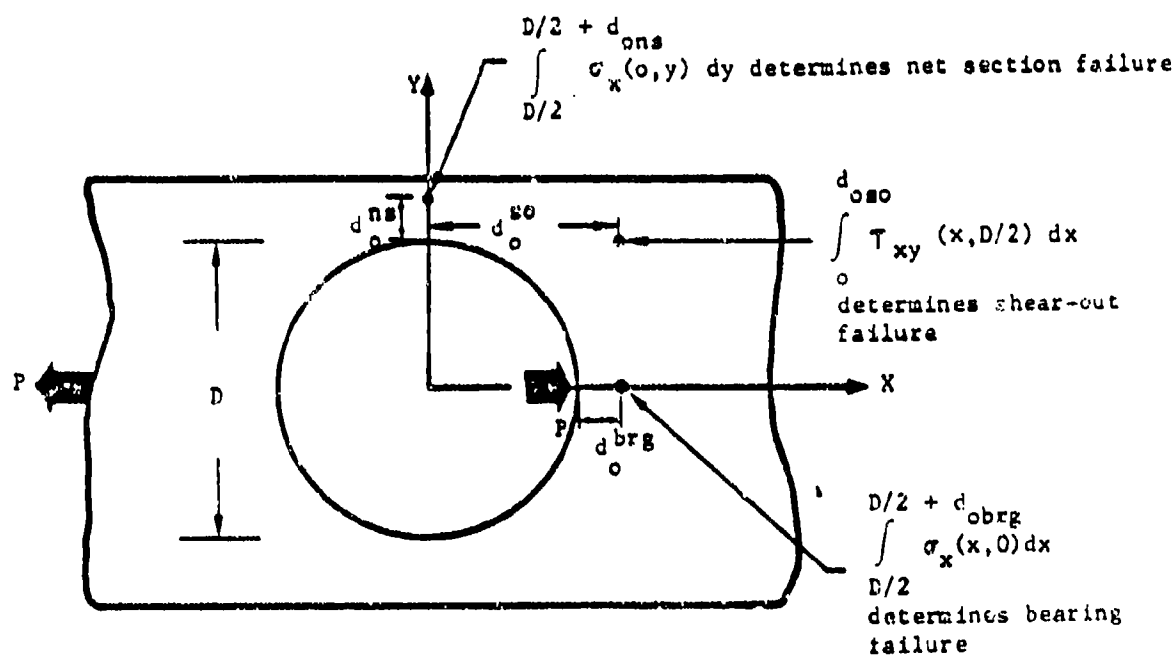


Figure 30. The Characteristic Distances Used in the Average Stress Failure Criteria.

predicted. When τ_{xy} (a_{oso} , $D/2$) exceeds the unnotched ply shear strength, a shear-out mode of ply failure is predicted. The average stress failure criterion predicts these failures based on averaged values of the mentioned stress components over selected characteristic distances (d_{ons} , d_{oso} , and d_{obrg}) that are larger in magnitude compared to those used in conjunction with the point stress criterion (see Figure 30).

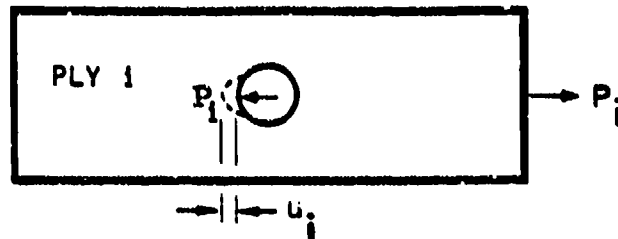
Of the three ply failure modes, only the net section mode causes the ply to become almost ineffective (total failure). The bearing mode of failure causes the ply to suffer a reduction in its effective modulus without losing its load-carrying capacity. The shear-out mode of failure causes a ply to become ineffective only when it is delaminated from the adjacent plies. When a ply suffers any of the above failures, its load versus deflection response is at the knee of the bilinear representation in Figure 31. The damaged ply can carry additional load until total ply failure is precipitated. The SASCJ computer code automatically stores the damage state in every ply in the bolted plates, and reassigns values for ply moduli to appropriately represent predicted ply failures. When a ply suffers total failure, its modulus is set equal to zero, and the redistribution of the joint load among the remaining plies is computed. A typical overall load versus deflection behavior of the joint is shown in Figure 32, indicating the effects of local and total ply failures.

The maximum strain (fiber directional), Hoffman and Tsai-Hill criteria are applied along a path that is concentric to the fastener hole, at a characteristic distance (a_o) from the hole boundary (see Figure 33). The location along this path where the selected criterion is satisfied determines the failure location. The maximum strain criterion predicts fiber failure in a ply (total ply failure) when its fiber directional strain exceeds the failure values (ϵ_{11}^{tu} or ϵ_{11}^{cu}).

The Hoffman failure criterion, based on inplane ply

$$K_2 = aK_1$$

$$P_{\text{ULTIMATE}} = \beta P_{\text{INITIAL}}$$



$$q_i = P_i / h_i$$

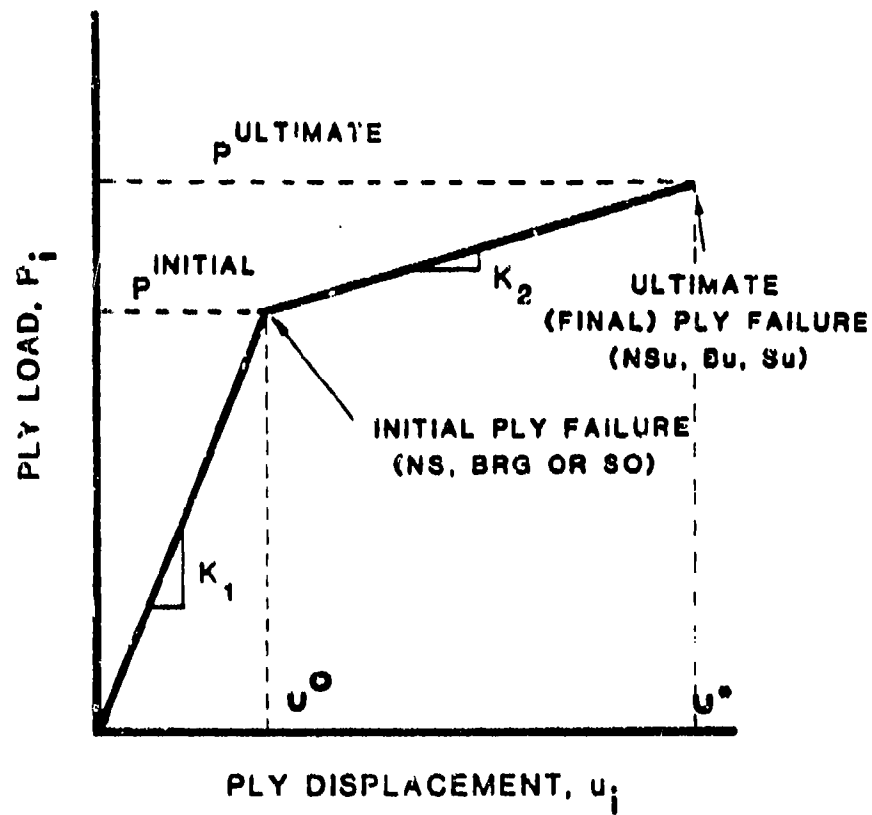


Figure 31. Bilinear Elastic Behavior of a Ply.

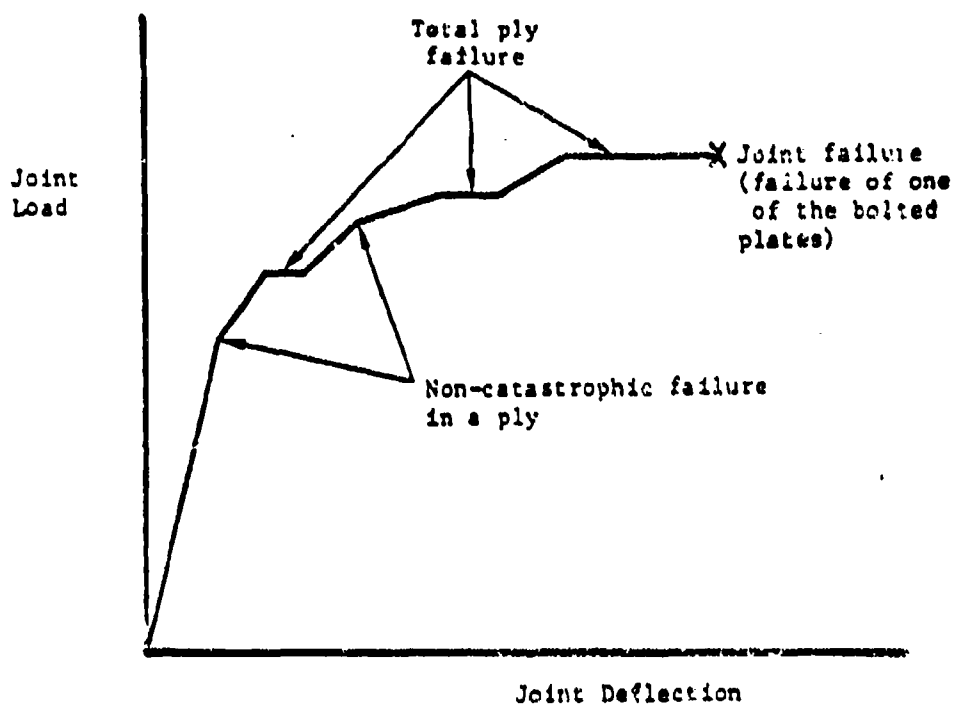


Figure 32. A Schematic Representation of the Overall Load Versus Deflection Response of the Joint.

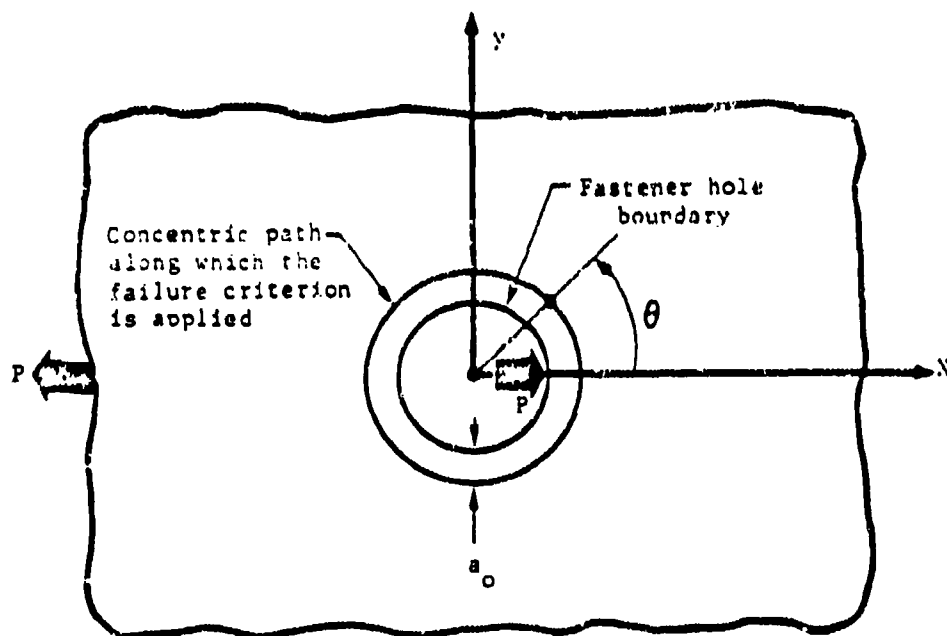


Figure 33. The Characteristic Distance (a_0) Defining the Region Where the Maximum Strain, Hoffman or Hill Criterion is Applied.

stresses, states that total ply failure will occur when the failure index (H) in the following equation reaches a value of unity:

$$\begin{aligned} & (\sigma_1^2 - \sigma_1\sigma_2)/X_cX_t + \sigma_1(X_c - X_t)/X_cX_t + \sigma_2^2/Y_cY_t + \sigma_2(Y_c - Y_t)/Y_cY_t + \\ & \sigma_6^2/S^2 = H \end{aligned}$$

In the above equation, σ_1 , σ_2 and σ_6 are the ply stresses in the fiber coordinate system, X_t and X_c are the uniaxial tensile and compressive material strengths along the fiber direction (1), Y_t and Y_c are the uniaxial tensile and compressive material strengths perpendicular to the fiber direction (2), and S is the material shear strength in the 1-2 plane.

In the SASCF code, the Hoffman criterion is applied along a path that is concentric to the fastener hole, defined by the characteristic distance a_0 (see Figure 33). At selected points along this path, the following expressions for the failure values of the ply load (P_f) are computed:

$$P_f = (-b \pm \sqrt{b^2 - 4ac})/2a$$

where

$$a = [(\sigma_1^2 - \sigma_1\sigma_2)/X_cX_t + \sigma_2^2/Y_cY_t + \sigma_6^2/S^2]/P_1^2$$

$$b = [(X_c - X_t)\sigma_1/X_cX_t + (Y_c - Y_t)\sigma_2/Y_cY_t]/P_1$$

$$c = -1, \text{ and}$$

$$P_1 = \text{ply load at which } \sigma_1, \sigma_2 \text{ and } \sigma_6 \text{ are computed}$$

The location where the smallest non-negative value for P_f is computed identifies the failure initiation point.

The Hoffman criterion predicts total ply failure and the failure location, but does not identify the mode of failure. The failure location, though, generally indicates the possible failure mode. Referring to Figure 33, if failure is predicted near $\theta=0^\circ$, a bearing mode of failure is suspected. If the failure location is near $\theta=90^\circ$, a net section mode of failure is suspected. And, intermediate values of θ indicate a shear-out mode of failure. The Tsai-Hill criterion can be obtained from the Hoffman criterion by setting $X_c = X_t$ and $Y_c = Y_t$. This criterion, therefore, does not account for different strengths under tension and compression. The ply failure load (P_f) in this case is computed to be $1/\sqrt{a}$.

3.1.4 Interlaminar Failure Criterion

Delamination between plies is predicted by incorporating computed shear strains at the interfacial locations into a maximum shear strain criterion. At the interface between plies i and j , for example, the shear strain is computed to be:

$$\gamma_{xz}^{i-j} = (u_i - u_j) / h_\alpha$$

where h_α is the ply thickness in the plate containing plies i and j . This expression for the shear strain is approximate. Plies i and j are assumed to delaminate when γ_{xz}^{i-j} exceeds a failure value. The failure value for γ_{xz} is determined by correlating predictions with observations for a sample test case.

3.2 SASCJ Input Description

SASCJ assumes a uniaxial tensile or compressive load to

be applied to a single fastener bolted joint, in a single or a double shear configuration (see Figures 34 and 35). The code requests information for a general bearing/by-pass situation. If the joint is a symmetric double shear configuration, only half the joint is analyzed (see Figure 35). For example, if plate 2 in Figure 35 is metallic, the input thickness should be half the actual value, and if plate 2 is a laminate, only the layup from the surface to its midplane should be input. The analysis accounts for the joint symmetry through appropriate symmetry conditions at the midplane location (see Figure 35).

A sample SASCJ problem is now presented to describe the input requirements for the code. It addresses a steel-to-composite joint in a single shear configuration (see Figure 34). The input is requested by SASCJ in an interactive mode. Figure 36 presents the code requests and the user replies for the sample joint. Though the information in Figure 36 is self-explanatory, a description of the input quantities is presented below.

The first input quantity specifies that the problem addresses a bearing/by-pass situation with a by-pass ratio of 0.99 --nearly an open hole situation. The second and third input quantities specify that a static tensile load is applied in a single shear configuration. Subsequently, the two bolted plates are specified to be either a composite laminate or a metal. If the bolted plate is a laminate, SASCJ requests the user to specify the number of plies in that plate (20). Note again that, for a double shear configuration, only half the thickness of the second plate should be defined (see Figure 35). SASCJ then requests the user to specify the thickness of the metallic plate (0.25). For the laminated plate, SASCJ requests, in sequence, the average cured ply thickness (0.006), the number of distinct ply orientations (4), definition of the four orientations (0.0, +45.0, -45.0 and 90.0), and the laminate stacking sequence -- $[(45/0/-45/0)_2/0/90]_c$. SASCJ automatically assumes a metallic plate to be divided into thirty identical layers. The number of layers in a laminate is controlled

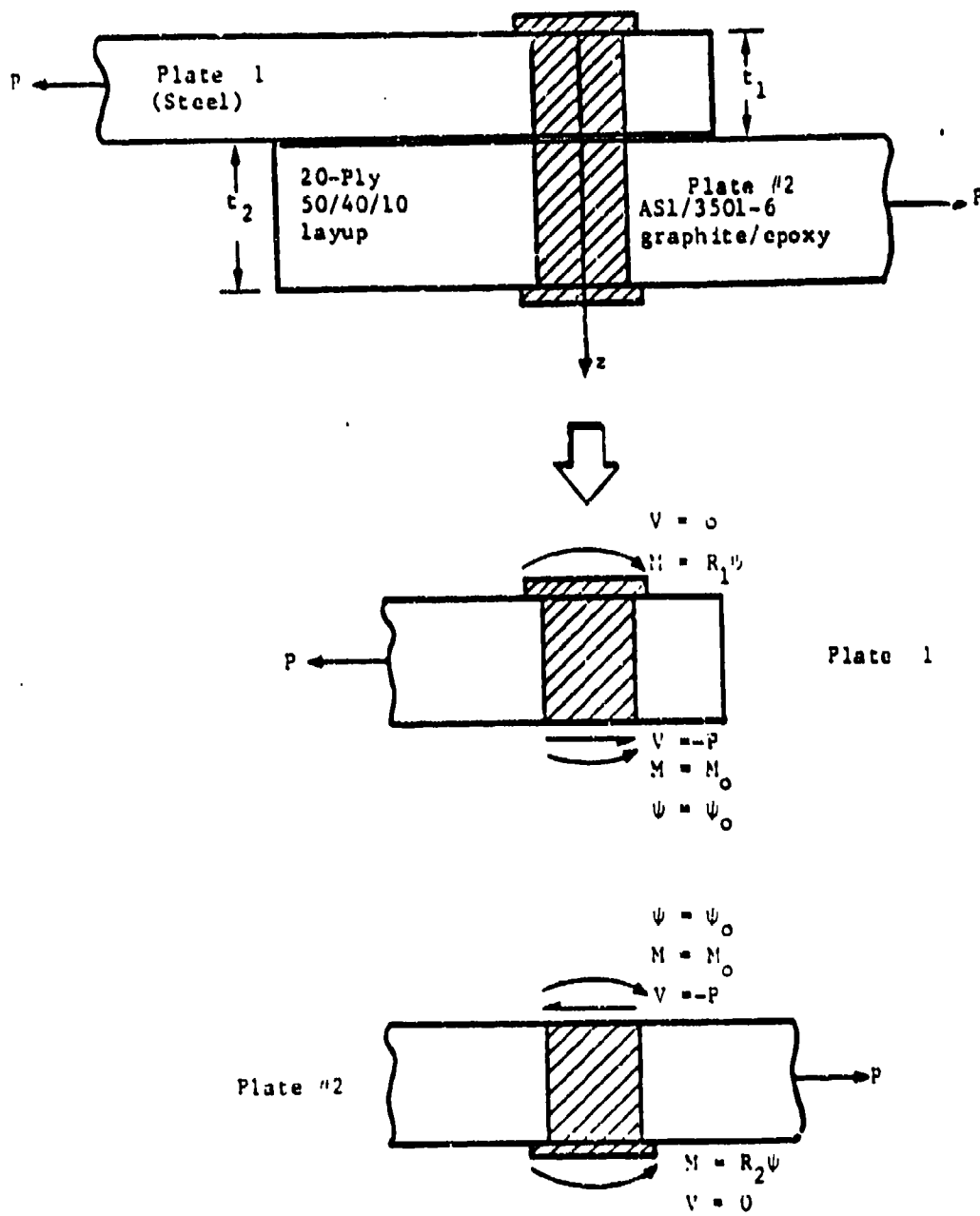


Figure 34. SASCJ Analysis of a Joint in Single Shear.

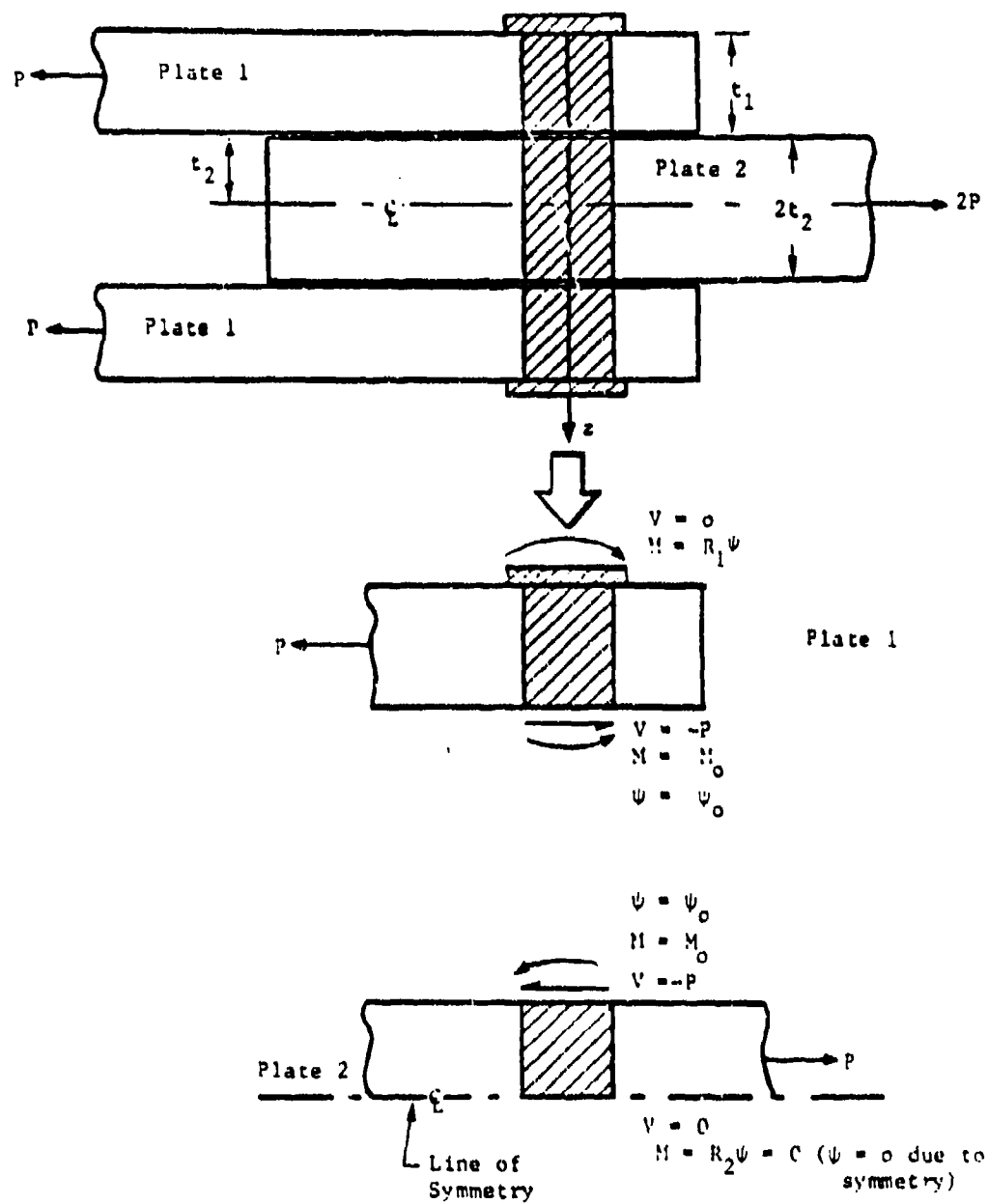


Figure 35. SASCJ Analysis of a Joint in Double Shear.


```

171500061 TEMPERATURE ASSUMED AS MEMBERTEMP
PROGRAM SASCIJ
PROGRAM SASCIJ PREDICTS FAILURE LOADS OF
MECHANICALLY FASTENED COMPOSITE LAP JOINTS.
SINGLE OR DOUBLE LAP SHEAR JOINTS.
PROGRAM ASSURES THAT INPUT PARAMETERS ARE
IN ENGLISH UNITS - LENGTHS ARE INPUT
IN INCHES AND MODULI AND STRENGTHS ARE
EXPRESSED IN PSI.
ENTER BYPASS RATIO ALPHA:
ALPHA=0 FOR FULL BEARING
ALPHA=1 FOR OPEN HOLE
0 (ALPHA=1 FOR GENERAL BYPASS
7
0.99
ENTER:
1 FOR STATIC TENSION
2 FOR STATIC COMPRESSION
7
1
ENTER:
1 FOR SLS (SINGLE LAP SHEAR)
2 FOR BLS (DOUBLE LAP SHEAR)
7
1
IS THE TOP PLATE A COMPOSITE OR A METAL ?
ENTER C OR M IN THE FIRST FIELD
7
C
INPUT MATERIAL DESCRIPTION OF THIS PLATE
EX: AGA/3501-6
aba01
IS THE BOTTOM PLATE A COMPOSITE OR A METAL ?
ENTER C OR M IN THE FIRST FIELD
7
C
INPUT MATERIAL DESCRIPTION OF THIS PLATE
EX: AGA/3501-6
ba1/3501-6 (105-0/-45/012/0/30)0
NOTE: FOR COMPUTATIONAL PURPOSES A
METALLIC PLATE IS REVELED AS P.30 PLY
LAMINATE OF 0 DEGREE PLYS WITH ISOTROPIC
MATERIAL PROPERTIES
NOTE: NUMERICAL DESIGNATIONS FOR THE
PLATES ARE:
TOP PLATE - NO 1
BOTTOM PLATE - NO 2
INPUT NUMBER OF PLYS IN PLATE NO 2
7
14 (N = 2)
20
INPUT THICKNESS OF PLATE NO 1
7
0.25
INPUT PLY THICKNESS IN PLATE NO 2
7
0.005
INPUT NUMBER OF DISTINCT PLY ORIENTATIONS
IN PLATE NO 2
7
4
FOR PLATE NUMBER 2 1
INPUT ORIENTATION OF PLY TYPE NO 1
7
0
INPUT ORIENTATION OF PLY TYPE NO 2
7
45.0
INPUT ORIENTATION OF PLY TYPE NO 3
7
-45.0
INPUT ORIENTATION OF PLY TYPE NO 4
7
90.0
INPUT TYPE OF PLY IN PLATE NO 2 FROM TOP
TO BOTTOM
PLY TYPE ORIENTATION
1 0.0 DEGREES
2 45.00 DEGREES
3 -45.00 DEGREES
4 90.00 DEGREES
INPUT TYPE OF PLY FOR PLY NO 1
7
1
INPUT TYPE OF PLY FOR PLY NO 2
7
2
INPUT TYPE OF PLY FOR PLY NO 3
7
1
INPUT TYPE OF PLY FOR PLY NO 4
7
2
INPUT TYPE OF PLY FOR PLY NO 5
7
1
INPUT TYPE OF PLY FOR PLY NO 6
7
2
INPUT TYPE OF PLY FOR PLY NO 7
7
1
INPUT TYPE OF PLY FOR PLY NO 8
7
2
INPUT TYPE OF PLY FOR PLY NO 9
7
1
INPUT TYPE OF PLY FOR PLY NO 10
7
2

```

Figure 36. Sample SASCIJ Input.

```

INPUT TYPE OF PLY FOR PLY NO 11
?
4
INPUT TYPE OF PLY FOR PLY NO 12
?
1
INPUT TYPE OF PLY FOR PLY NO 13
?
1
INPUT TYPE OF PLY FOR PLY NO 14
?
2
INPUT TYPE OF PLY FOR PLY NO 15
?
1
INPUT TYPE OF PLY FOR PLY NO 16
?
2
INPUT TYPE OF PLY FOR PLY NO 17
?
1
INPUT TYPE OF PLY FOR PLY NO 18
?
1
INPUT TYPE OF PLY FOR PLY NO 19
?
1
INPUT TYPE OF PLY FOR PLY NO 20
?
1
INPUT THE ENGINEERING PROPERTIES OF PLATE NO 1
INPUT YOUNG'S MODULUS AND POISSON'S RATIO
30.046 0.2
INPUT THE ENGINEERING PROPERTIES OF PLATE NO 2
INPUT YOUNG'S MODULUS, E1 AND E2
18.546 1.946
INPUT THE SHEAR MODULUS AND MAJOR POISSON'S RATIO
0.8546 0.1
INPUT MATERIAL DESCRIPTION FOR FASTENER
steel
INPUT YOUNG'S MODULUS AND POISSON'S RATIO FOR
THE FASTENER
30.046 0.3
INPUT THE DIAMETER OF THE FASTENER
0.3185
FASTENER TYPE
ENTER 1 FOR PROTRUDING HEAD
?
2 FOR COUNTERSUNK HEAD
?
1
PLATE PROPERTIES ARE SPECIFIED BY
INPUTTING THE COORDINATES OF THE CORNER
VERTICES. NOTE: THE ORIGIN IS AT THE FASTENER
CENTER; INPUT COORDINATES ACCORDINGLY
U3 U2
MOLE
U3 U2
U4 U1
CENTROID
U4 U1
APPLIED LOCAL CONCENTRATIONS
FOR PLATE NO 1 (TOP) NORMAL LOADS ARE APPLIED
BETWEEN U3 AND U4
FOR PLATE NO 2 (BOTTOM) NORMAL LOADS ARE APPLIED
BETWEEN U1 AND U2
FOR PLATE NUMBER 1:
ENTER X,Y COORDINATES OF U 1
3.0 -0.9375
ENTER X,Y COORDINATES OF U 2
2.0 0.9375
ENTER X,Y COORDINATES OF U 3
-3.0 0.9375
ENTER X,Y COORDINATES OF U 4
-3.0 -0.9375
FOR PLATE NUMBER 2:
ENTER X,Y COORDINATES OF U 1
3.0 -0.9375
ENTER X,Y COORDINATES OF U 2
3.0 0.9375
ENTER X,Y COORDINATES OF U 3
-3.0 0.9375
ENTER X,Y COORDINATES OF U 4
-3.0 -0.9375
SELECT FAILURE CRITERIA:
ENTER 1 FOR POINT STRESS CRITERION
?
ENTER 2 FOR MAXIMUM STRESS CRITERION
?
2
AVERAGE STRESS CRITERION
NO IS THE CHARACTERISTIC DISTANCE OVER WHICH
STRESSES ARE AVERAGED AND COMPARED WITH SPECIFIED
STRESSING TO PREDICT FAILURE
ENTER AN UNLESS OTHERWISE NOTED TO THE THREE
PLY FAILURE MODES IN PLATE NO
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

```

Figure 36. Sample SASCI Input
(Cont Inued) .

INPUT BETA1 FOR NET SECTION ULTIMATE
 BETA2 FOR BEARING ULTIMATE
 BETA3 FOR SHEAROUT ULTIMATE
 1.02 1.5 1.12

TO AVOID LENGTHY RUN TIMES DUE TO
 STRESS FIELD RECOMPUTATION SPECIFY THE
 NUMBER OF ULTIMATE PLY FAILURES AFTER
 WHICH JOINT FAILURE WILL BE PREDICTED
 ENTER: NO OF ULTIMATE FAILURES
 ?

24
 ENTER NO VALUES CORRESPONDING TO THE THREE
 PLY FAILURE MODES IN PLATE NO 2

MODE - NET SECTION
 A0M - BEARING
 A0S - SHEAR OUT
 INPUT A0M1, A0M2, A0M3

? 0.1 0.025 0.08

FOR PLATE NUMBER 1 ENTER THE THREE STRENGTHS
 REQUIRED TO PREDICT THE THREE FAILURE MODES
 FST-UNNOTCHED STRENGTH IN TENSION
 FSC-UNNOTCHED STRENGTH IN COMPRESSION
 FSO-UNNOTCHED STRENGTH IN SHEAR-OUT

INPUT FST,FSC,F50

? 250.000 300.000 200.000

FOR PLATE NO 2 ENTER FIBER ULTIMATE
 STRAIN VALUES

EPSILON ULT IN COMPRESSION
 EPSILON ULT IN TENSION
 GAMMA ULT IN SHEAR

(UNITS: IN/IN)
 ?

0.014 0.011 0.012
 SASCJ ASSURES A BILINEAR PLY BEHAVIOR. THE
 INITIAL MODULUS, E1, IS COMPUTED BY THE CODE.
 THE REDUCED MODULUS, E2, FOR INITIAL FAILURE
 IN NET SECTION, SHEAROUT OR BEARING IS COMPUTED
 BY THE FORMULA $E2 = \alpha \cdot E1$.

FOR PLATE NUMBER 1 INPUT ALPHA VALUES FOR
 NET SECTION, SHEAROUT AND BEARING FAILURE
 ?

001 0.1 0.1
 INPUT SCALE FACTORS FOR P ULTIMATE
 CALCULATION SUCH THAT $P1(ULT) = BETA1(INITIAL)$
 INPUT BETA1 FOR NET SECTION ULTIMATE
 BETA2 FOR BEARING ULTIMATE
 BETA3 FOR SHEAROUT ULTIMATE

? 1.02 1.5 1.12
 SASCJ ASSURES A BILINEAR PLY BEHAVIOR. THE
 INITIAL MODULUS, E1, IS COMPUTED BY THE CODE.
 THE REDUCED MODULUS, E2, FOR INITIAL FAILURE
 IN NET SECTION, SHEAROUT OR BEARING IS COMPUTED
 BY THE FORMULA $E2 = \alpha \cdot E1$.

FOR PLATE NUMBER 2 INPUT ALPHA VALUES FOR
 NET SECTION, SHEAROUT AND BEARING FAILURE
 ?

0.1 0.1 0.1
 INPUT SCALE FACTORS FOR P ULTIMATE
 CALCULATION SUCH THAT $P1(ULT) = BETA1(INITIAL)$

Figure 36. Sample SASCJ Input
 (Concluded).

by the user. In Figure 36, each physical ply is modeled as one layer. For this sample problem, for example, the user could also specify each physical ply to be divided into two identical plies, by setting the number of plies in the laminate to be 40, the cured ply thickness to be 0.003 inch, and repeating each ply orientation in the stacking sequence twice.

Subsequent to the above information, SASCJ requests the material properties for plates 1 and 2 (Young's modulus and Poisson's ratio for a metal, and Young's modulus, shear modulus and the major Poisson's ratio for each lamina, in the fiber coordinate system). The fastener modulus, Poisson's ratio, diameter and head type (protruding head or countersunk) are requested next. Following that, the geometry of the bolted plates is defined by specifying the coordinates for the plate corners, assuming that the origin is located at the center of the fastener hole.

The last block of data addresses the selected failure criterion and the corresponding failure parameters. In the sample problem in Figure 36, the average stress failure criteria are selected for failure prediction (4). The characteristic distances for net section, bearing and shear-out modes of failure are then specified for the two plates. This is followed by the unnotched strengths for the two plates under tension, compression and inplane shear. Next, SASCJ requests the parameters that define the bilinear material behavior. These are the factors that define the modulus change after initial failure, and the ratio of the ultimate ply failure load to the initial ply failure load. Different factors may be specified for the three failure modes. Finally, the approximate ultimate shear strain value is requested for delamination prediction. A large value is generally specified for a metallic plate, to preclude the prediction of delaminations that are not applicable to these materials.

3.3 SASCJ Output Description

For the sample problem defined in Figure 36, SASCH provides the output shown in Figure 37. The input data for the bolted plates is initially reproduced for user verification. Subsequently, the sequence of failures in the bolted laminate and the corresponding joint load levels are printed. Note that the ultimate failure of a ply (shear-out of the 45 degree plies) does not necessarily imply joint failure. In the considered sample problem, shear-out of the 0 degree plies limits the load-carrying capacity of the joint. Every ply suffers a two-stage failure as described before (Figure 31).

When executed in some systems, SASCH could yield underflow messages after many plies have suffered total failure. This may occur when the double precision format is not followed in entering input data. Nevertheless, the user is advised to ignore these messages.

3.4 Description of SAMCH Analysis

This section presents an overview of the strength analysis in the SAMCH computer code, a description of the developed special finite elements, and the analytical procedure used in SAMCH to predict fastener loads, the critical fastener or cut-out location, the corresponding joint strength and the failure mode.

A flow chart of SAMCH operations is presented in Figure 38. As input, SAMCH requires the user to specify how the bolted plates are divided into plain elements and elements with loaded or unloaded holes. The bolted plates are currently assumed by SAMCH to be subjected to uniaxial tensile or compressive loading, in a single or double shear configuration. Additional input requirements for the SAMCH code include the material properties of the bolted plates and fasteners, and the fastener size, location and torque. The material properties of the bolted laminates include the tensile and compressive failure strains in the fiber direction of the lamina, and the characteristic distances over which stresses are averaged to

PROGRAM SASCI

A SINGLE LAP SHEAR JOINT WILL BE ANALYZED

WITH A PARTIALLY LOADED HOLE

STRESS RATIO = 0.5000+00

LOADED IN STATIC TENSION

PLATE NO 1 1

STEEL

T = 0.2500+00 INCHES

MATERIAL PROPERTIES

E1 = 0.3000+08 PSI

E2 = 0.3000+08 PSI

G12 = 0.1150+08 PSI

MU12 = 0.3000+00

MU21 = 0.3000+00

PLATE NO 2 1

AS1/3501-5 (145/8/-45/012/0/00)S

T = 0.1200+00 INCHES

MATERIAL PROPERTIES

E1 = 0.1850+08 PSI

E2 = 0.1850+07 PSI

G12 = 0.8500+06 PSI

MU12 = 0.3000+00

MU21 = 0.3000+01

FASTENER:

STEEL

DIAMETER = 0.3130+00 INCHES

MATERIAL PROPERTIES

E = 0.3000+08 PSI

MU = 0.3000+00

FAILURE ANALYSIS

AN AVERAGE STRESS CRITERION WILL BE USED

PLATE NUMBER 1

LAMINATE STRENGTH

NET SECTION ULTIMATE (TENS) = 0.2500+06 PSI
NET SECTION ULTIMATE (COMP) = 0.3000+06 PSI
PLATE 1: ULTIMATE = 0.3000+06 PSI
SHEAROUT ULTIMATE = 0.2000+06 PSI

CHARACTERISTIC DISTANCES

ADMT = 0.2000+00 INCHES

ROBR = 0.4000+00 INCHES

ROSO = 0.5000+00 INCHES

PLATE NUMBER 2

LAMINATE STRENGTH

NET SECTION ULTIMATE (TENS) = 0.1200+06 PSI

NET SECTION ULTIMATE (COMP) = 0.1500+06 PSI

BEARING ULTIMATE = 0.1500+06 PSI

SHEAROUT ULTIMATE = 0.1000+06 PSI

CHARACTERISTIC DISTANCES

ADMT = 0.1000+00 INCHES

ROBR = 0.2500+01 INCHES

ROSO = 0.8000+01 INCHES

GEOMETRY OF PLATE NO 1 1

COORDINATES OF CORNER VERTICES

-3.000, 0.000 3.000, 0.000

-3.000, -0.000 3.000, -0.000

FASTENER HOLE DIAMETER = 0.3130+00 INCHES

E/D RATIO = 0.0000+01

U/D RATIO = 0.0000+01

GEOMETRY OF PLATE NO 2 1

COORDINATES OF CORNER VERTICES

Figure 37. SASCI Output for the Problem Defined in Figure 36.

1.000, 0.938 3.000, 0.538
 3.000, 0.938 3.000, -0.238
 FASTER HOLE DIAMETER = 0.3130+00 INCHES
 E-L RATIO = 0.9500+01
 U-D RATIO = 0.6000+01

T-4 PREDICTED JOINT FAILURE
 LOAD IS 0.486270+00 LBS

READY

FAILURE MODE ABBREVIATIONS:

- ND - NO ADDITIONAL DAMAGE AT CURRENT JOINT LOAD
- DL - DELAMINATION
- SO - SKEW-OUT
- BR - BEARING
- MS - ME - SECTION
- SUO - ULTIMATE FAILURE AFTER SO AND DL
- SU - ULTIMATE FAILURE IN SO
- BU - ULTIMATE FAILURE IN BR
- MSU - ULTIMATE FAILURE IN MS
- ULT - ULTIMATE FAILURE

INCREMENT NO	JOINT LOAD	MODE	PLV TYPE	MODE
1	0.8500+04	32	0.0	MS
2	0.8700+04	34	0.0	MS
3	0.8700+04	36	0.0	MS
4	0.8710+04	38	0.0	MS
5	0.8720+04	38	0.0	MS
6	0.8730+04	42	0.0	MS
7	0.8730+04	43	0.0	MS
8	0.8730+04	45	0.0	MS
9	0.8730+04	47	0.0	MS
10	0.8740+04	49	0.0	MS
11	0.8760+04	32	0.0	MSU
12	0.8170+04	34	0.0	MSU
13	0.7470+04	36	0.0	MSU
14	0.6700+04	38	0.0	MSU
15	0.5340+04	39	0.0	MSU
16	0.5340+04	42	0.0	MSU
17	0.4610+04	43	0.0	MSU
18	0.3890+04	45	0.0	MSU
19	0.3150+04	47	0.0	MSU
20	0.2450+04	49	0.0	MSU
21	0.2110+04	71	45.000	MS
22	0.2110+04	75	45.000	MS
23	0.4320+04	48	45.000	MS
24	0.4320+04	50	45.000	MS
25	0.4320+04	33	-45.000	MS
26	0.4320+04	37	-45.000	MS
27	0.4320+04	44	-45.000	MS

Figure 37. SASCJ Output for the Problem Defined in Figure 36 (Concluded).

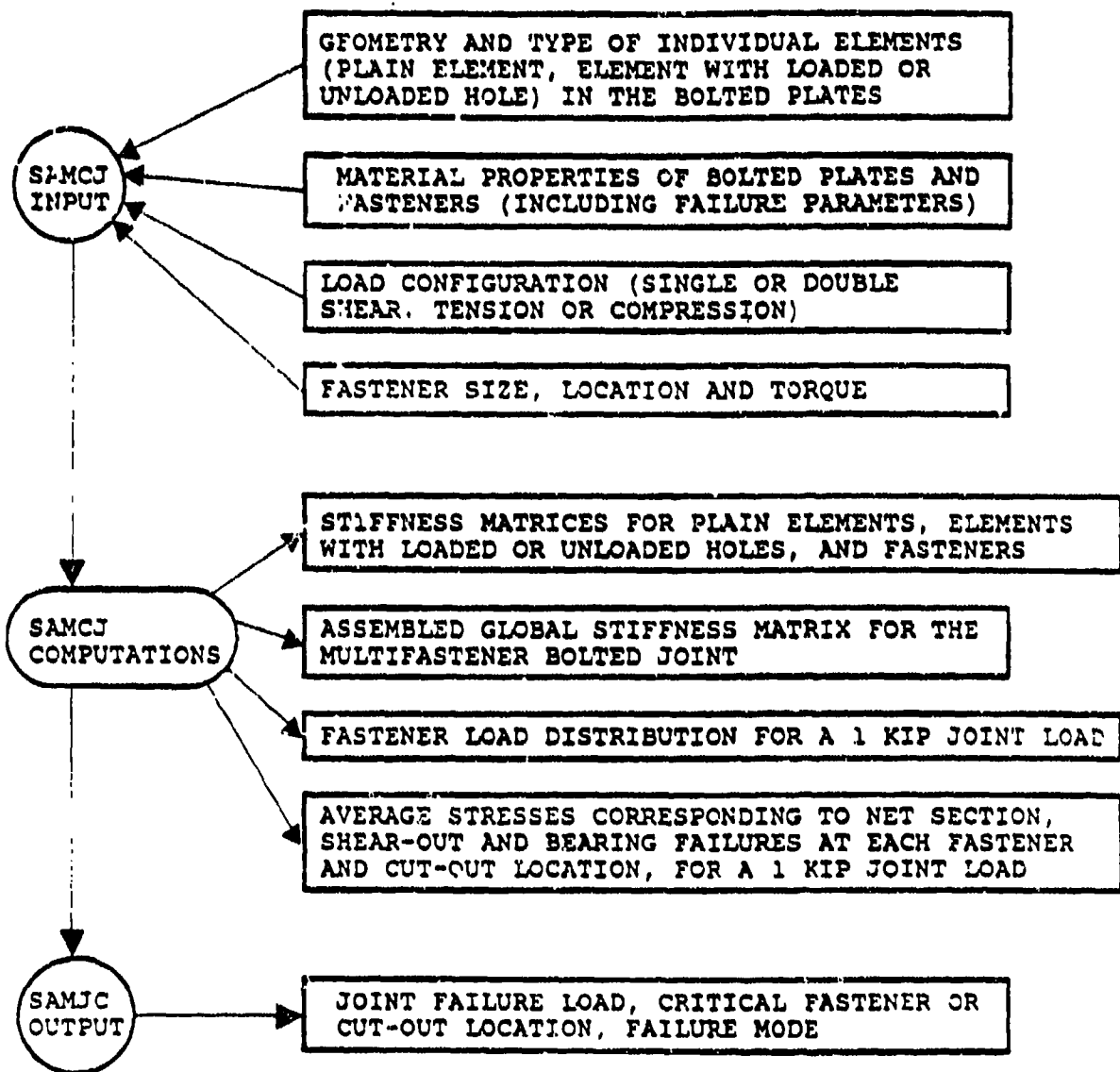


Figure 38. Flow Chart of SAMCJ Operations.

predict net section, shear-out and bearing failures at the fastener or cut-out location.

With the above input, SAMCJ performs the following computations. It initially generates stiffness matrices for all the special finite elements, namely, plain elements, elements with loaded or unloaded holes, and effective fastener elements (see Reference 7). The individual stiffness matrices are subsequently assembled to obtain the global stiffness matrix for the bolted joint. A 1 kip uniaxial tensile or compressive joint load is imposed on the left end of the top plate, in accordance with the input instructions (see Figure 30). The nodes at the right end of the bottom plate are constrained from translating in the load direction, and one of these nodes is also constrained in the transverse direction, to preclude all rigid body translations. The solution to this finite element formulation of the bolted joint provides the axial and transverse components of the load at every fastener location, corresponding to a 1 kip joint load. Also computed are the average net section, shear-out and bearing stresses at every fastener and cut-out location, corresponding to a 1 kip joint load.

SAMCJ provides, as output, the failure value of the uniaxial joint load, the critical fastener or cut-out location, and the joint failure mode. These are obtained as follows. The tensile, compressive and shear strengths of the plain laminates are computed based on the input tensile and compressive failure strains in the fiber direction of the lamina. The ratios of the averaged stresses to the corresponding unnotched laminate strengths, at selected locations around each fastener and cut-out boundary, are compared to predict the failure mode, the critical fastener or cut-out location and the joint failure load. SAMCJ predicts net section, shear-out and bearing modes of failure at the laminate level. In the SASCI code, similar failure predictions for single fastener joints in composites are made at the lamina level. Consequently, the failure parameters (characteristic distances for

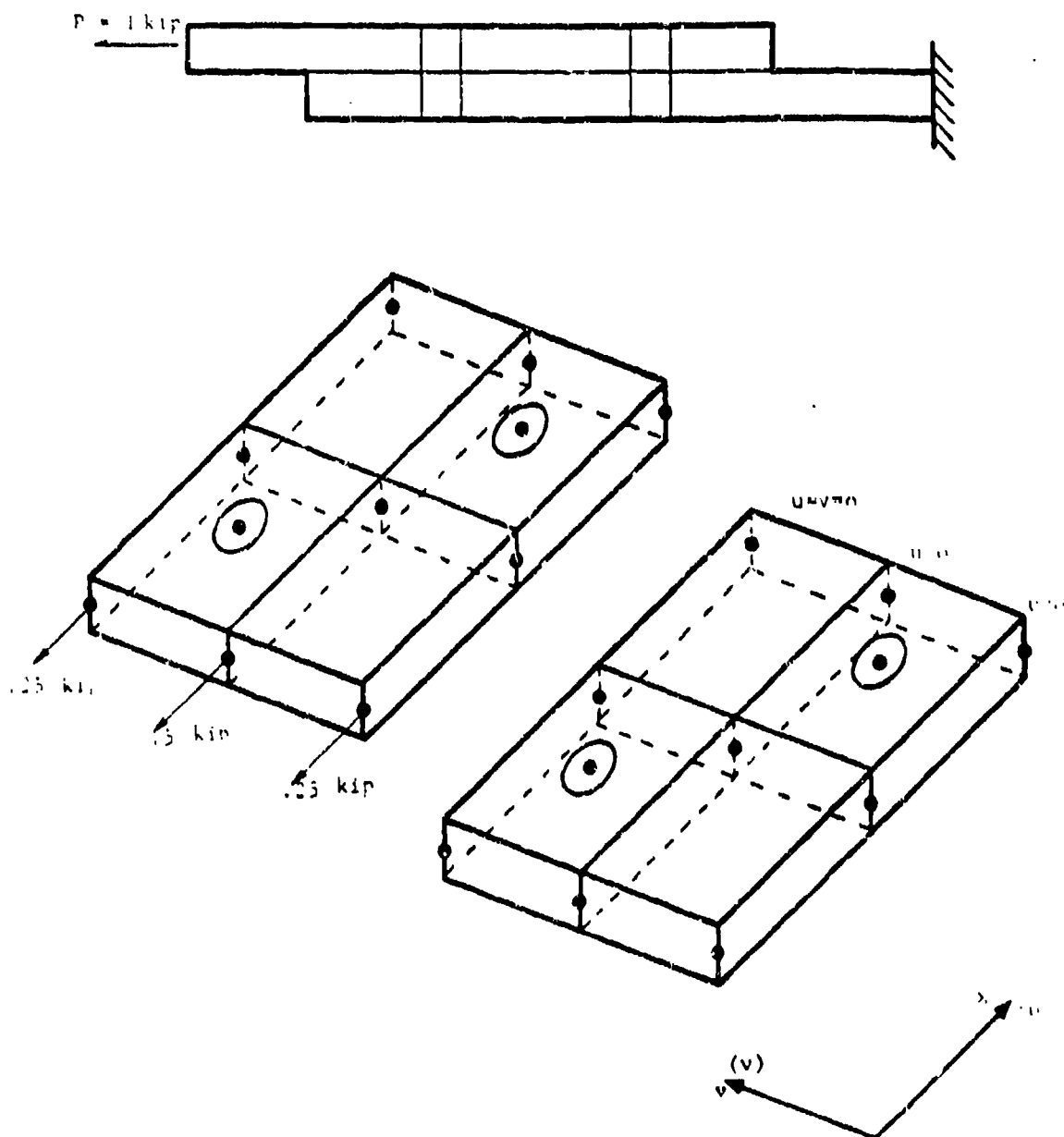


Figure 39. Application of Load and Displacement Boundary Conditions in the SAMCJ Code.

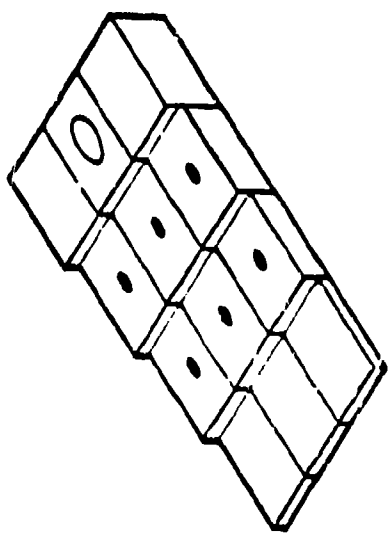
the three failure modes) used with SAMCJ are different from those used with SASCJ.

The incorporation of the transverse effective fastener stiffness values provides SAMCJ the capability to account for fastener flexibility, torque, and load eccentricity (single versus double shear load transfer). The FDFA code, developed in Reference 6, is used to compute the effective fastener transverse stiffnesses, along and perpendicular to the load direction (see Reference 7). The effect of the laminate stacking sequence is also accounted for in this analysis. SAMCJ executes FDFA twice to account for the layup variation (by 90 degrees) from the loading direction to the perpendicular direction.

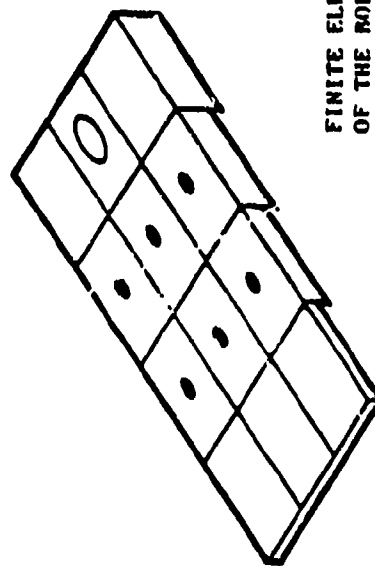
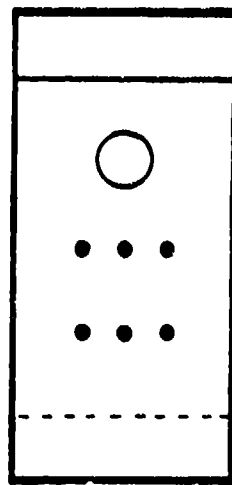
SAMCJ accounts for stress concentration interaction effects introduced by neighboring cut-outs, free edges and proximate fastener locations. This is made possible by the use of the FIGEOM stress analysis, developed in Reference 6, to generate element stiffness matrices (see Reference 7). FIGEOM accounts for finite planform plate dimensions through a boundary collocation solution procedure (see Reference 6).

SAMCJ computes the magnitude and the orientation of the load at each fastener location. It is a two-dimensional load distribution analysis that does not rely on an experimental measurement of "joint stiffness." In a design situation, many fastener arrangements can be analytically and economically evaluated by SAMCJ to arrive at the best fastener pattern for the assumed loading conditions.

When the bolted plates are tapered, the SAMCJ user can input equivalent uniform thickness elements to approximate the tapering effect (see Figure 40). Adjacent elements in the tapered plate will have different thickness values. This feature is essential in the analysis of practical structural joints.



SAMPLE TAPERED JOINT



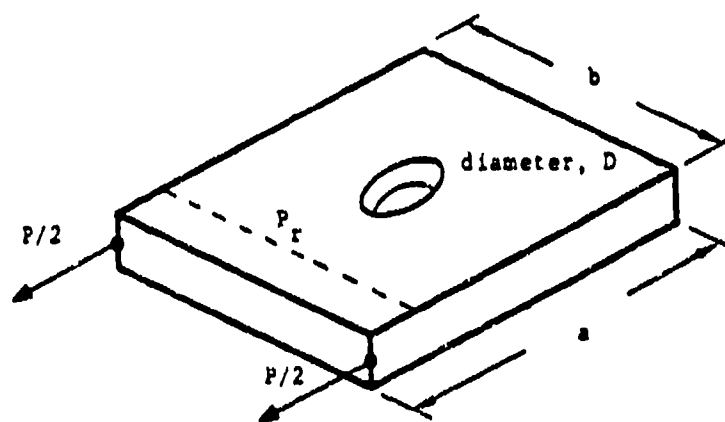
FINITE ELEMENT MODEL
OF THE BOLTED PLATES

Figure 40. Finite Element Model of a Sample Tapered Bolted Joint.

SAMCJ has been developed for the strength prediction of bolted laminated structural parts. It currently assumes that the selected fasteners preclude fastener failure. Also, it applies the same failure procedure to both the bolted plates, accounting for net section, shear-out and bearing failures via the averaged stress failure criteria applied at the laminate level. Joint failure is assumed to be a one-step (catastrophic) process. The strength of a bolted plate corresponds to the initial failure at a fastener or a cut-out location, in the bearing, shear-out or net section failure.

The unnotched laminate strengths, under tension, compression and inplane shear, are computed by SAMCJ based on input fiber-directional failure strain values (tensile and compressive). Laminate strengths under N_x and N_{xy} loadings (inplane normal and shear stress resultants, respectively) are assumed to correspond to first fiber failure in a ply. This simplistic strength prediction procedure introduces inaccuracies that have been acknowledged and discussed in the literature. Nevertheless, SAMCJ adopts this procedure for lack of a validated alternative.

Despite its versatility, SAMCJ has limitations that the user should be aware of. Reference 7 discusses the limitations of the five-noded (10 degrees of freedom) loaded hole element and the four-noded (8 degrees of freedom) unloaded hole element. In addition, when dividing a bolted plate into many elements (loaded or unloaded hole elements, as well as plain elements), it is advisable to maintain element geometries that do not render the generated stiffness matrices inaccurate. Figure 41 presents results from a study conducted on a singly-fastened metallic plate. P_r is the recovered load that is obtained by integrating the stresses along a line transverse to the load direction as shown in Figure 41. P is the applied load or the sum of the nodal loads (especially in the interior elements in a general multifastened plate). The recovered load (P_r) approaches the applied load value (P) when the plate aspect ratio (a/b) increases beyond unity, and when a/D and b/D have a minimum value of approximately three. In predicting failure in



$t = 0.3125 \text{ inch}$

a/D	b/D	P_r/P
1.6	1.6	5.38
3.2	1.6	2.27
6.4	1.6	1.57
16.0	1.6	1.29
1.6	3.2	1.24
3.2	3.2	1.76
6.4	3.2	1.37
16.0	3.2	1.16
1.6	6.4	-0.0995
3.2	6.4	0.989
6.4	6.4	1.23
16.0	6.4	1.16
3.2	16.0	-0.46
6.4	16.0	0.029
16.0	16.0	1.23

Figure 41. Element Load Recovery for Various a/D and b/D Ratios.

the net section, bearing and shear-out modes, the computed average stress values are multiplied by P/P_r , to remove geometry (modeling) effects from the computed stresses.

3.5 SAMCJ Input Description

To familiarize the user with SAMCJ input requirements, a sample problem is presented here (see Figure 42). The sample problem considers a six fastener composite-to-metal joint, with a one inch diameter circular cut-out adjacent to the first row of fasteners. Figure 42 presents the assumed nine element model of each of the two bolted plates, analyzed by SAMCJ. Figure 43 presents SAMCJ requests and user input in response to these requests, for the sample problem in Figure 42.

Though self-explanatory, the interactively entered SAMCJ input in Figure 43, for the sample problem in Figure 42, is described here for completeness. The first entry (1) identifies the loading configuration to be a single shear configuration. The second entry (1) identifies the load to be in static tension. The next two entries say that the top plate is a metal (M), identified as "Aluminum." The two entries following these say that the bottom plate is a composite laminate (C), identified as follows: "(45/0/-45/0)2/0/90)2s." Subsequently, the Young's modulus (10.0D6) and Poisson's ratio (0.3) for aluminum, and the fiber-directional, transverse and shear moduli and Poisson's ratio (18.5D6, 0.85D6 and 0.3, respectively) for the composite lamina are input. The next five entries specify that four (4) different fiber orientations are present in the laminate (0, 45, -45 and 90 degrees with respect to the loading direction). The following three entries say that the elements in the bottom plate contain one (1) layup of forty (40) plies, of 0.006175 inch thickness each. The stacking sequence for this layup is input next, where 1, 2, 3 and 4 refer to 0, 45, -45 and 90 degree fiber orientations, respectively. Subsequently, the fastener is identified as "Steel," and its Young's modulus, Poisson's ratio, and head type (30.0D6, 0.3, 0.3125 and

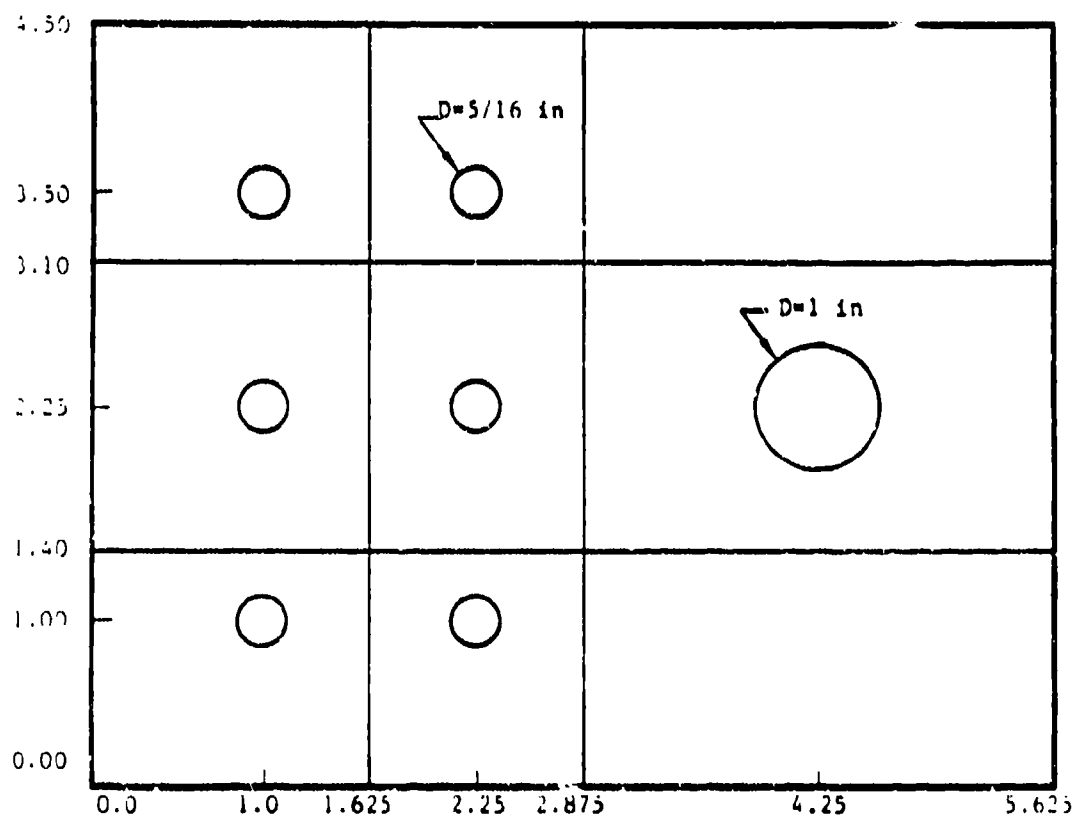
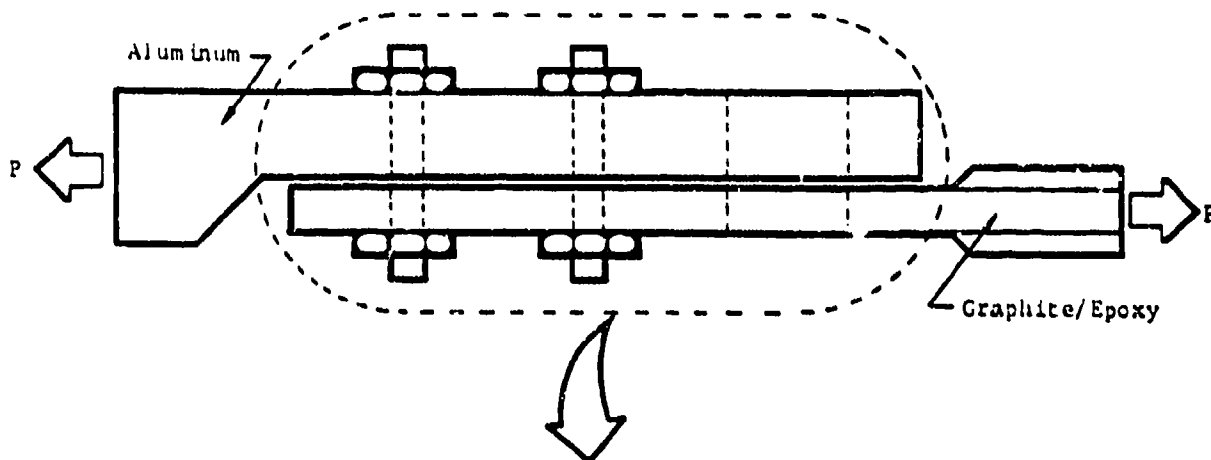


Figure 42. Nine Element Model of Each of the two Bolted Plates in the Sample Joint.

CALL NAME
1.50061 TEMPERATURE ASSUMED AS "TEMPERATURE"

PROGRAM SANCJ

PROGRAM SANCJ PREDICTS THE FAILURE LOAD, FAILURE LOCATION, AND FAILURE MODE IN MULTILAYER-FASTENED, SINGLE OR DOUBLE LAP COMPOSITE SHEAR JOINTS.

THE ANALYSIS ASSUMES THAT INPUT PARAMETERS ARE SPECIFIED IN ENGLISH UNITS - LENGTH IN INCHES, MODULI AND STRENGTHS IN PSI.

ENTER:
1 FOR SLS (SINGLE LAP SHEAR)
2 FOR DLS (DOUBLE LAP SHEAR)

1
1
ENTER:
1 FOR STATIC TENSION
2 FOR STATIC COMPRESSION

1
1
IS THE TOP PLATE A COMPOSITE OR A METAL?
ENTER C OR M IN THE FIRST FIELD

0
INPUT MATERIAL DESCRIPTION OF THIS PLATE
EX: ASA/3501-6
01000000
IS THE BOTTOM PLATE A COMPOSITE OR A METAL?
ENTER C OR M IN THE FIRST FIELD

6
INPUT MATERIAL DESCRIPTION OF THIS PLATE
EX: ASA/3501-6
1:45/0/-45/012/0/0010

NOTE: FOR COMPUTATIONAL PURPOSES A METALLIC PLATE IS MODELED AS A 30 PLY LAMINATE OF 0 DEGREE PLYS WITH ISOTROPIC MATERIAL PROPERTIES

INPUT THE ENGINEERING PROPERTIES OF THE TOP PLATE
INPUT YOUNG'S MODULUS AND POISSON'S RATIO

10.046 0.3
INPUT THE ENGINEERING PROPERTIES OF THE BOTTOM PLATE
INPUT YOUNG'S MODULUS, E1 AND E2

18.046 1.046
INPUT THE SHEAR MODULUS AND MAJOR POISSON'S RATIO

9.046 0.3
INPUT TOTAL NUMBER OF DISTINCT PLY ORIENTATIONS IN THE BOTTOM PLATE

1
INPUT ORIENTATION OF PLY TYPE NO 1
0.0

INPUT ORIENTATION OF PLY TYPE NO 2
45.0
INPUT ORIENTATION OF PLY TYPE NO 3
-45.0
INPUT ORIENTATION OF PLY TYPE NO 4
90.0

THICKNESS UNIFORMITIES MAY BE APPROXIMATED BY ASSIGNING DIFFERENT LAYUP TO ELEMENTS IN A COMPOSITE PLATE OR BY SPECIFYING DIFFERENT THICKNESSES TO ELEMENTS IN A METALLIC PLATE

ENTER NUMBER OF DIFFERENT LAYUPS IN THE BOTTOM PLATE

1
1
ENTER NUMBER OF PLYS IN LAYUP NO 1

40
ENTER PLY THICKNESS FOR THIS LAYUP

0.005175
ENTER SEQUENCE OF PLY TYPES FROM TOP TO BOTTOM

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007

```

0.3125
FASTENER TYPE
ENTER:
1 FINE PROUDING HEAD
2 FINE COUNTERSUNK HEAD

GRID LAYOUTS
ENTER NUMBER OF GRIDS IN TOP PLATE
7
22

ENTER 22 GRID POINTS
FORMAT: GRID ID, X AND Y COORDINATES
7
101 0.0 0.0
7
102 0.0 1.40
7
103 0.0 3.1
7
104 0.0 4.5
7
105 1.0 1.0
7
106 1.0 2.25
7
107 1.0 3.5
7
108 1.025 0.0
7
109 1.025 1.4
7
110 1.025 3.1
7
111 1.025 4.5
7
112 2.25 1.0
7
113 2.25 2.25
7
114 2.25 3.5
7
115 2.275 0.0
7
116 2.275 1.4
7
117 2.275 3.1
7
118 2.275 4.5
7
119 5.025 0.0
7
120 5.025 1.4

FASTENER DESCRIPTION:
INPUT MATERIAL DESCRIPTION FOR FASTENER
Steel
INPUT YOUNG'S MODULUS AND POISSON'S RATIO FOR
THE FASTENER
7
20.00E 0.3
INPUT THE DIAMETER OF THE FASTENER
7

```

Figure 43. SAMCJ Input for the Sample Problem in Figure 42 (Continued).

```

7 121 5.625 3.1
7 122 5.625 4.5
7 ENTER NUMBER OF CPIDS IN BOTTOM PLATE
82
7 ENTER 22 GRID POINTS
7
7 FORMAT: GRID ID, X AND Y COORDINATES
7
7 201 0.0 0.0
7 202 0.0 1.4
7 203 0.0 3.1
7 204 0.0 4.5
7 205 1.0 1.0
7 206 1.0 2.25
7 207 1.0 3.5
7 208 1.625 0.0
7 209 1.625 1.4
7 210 1.625 3.1
7 211 1.625 4.5
7 212 2.25 1.0
7 213 2.25 2.25
7 214 2.25 3.5
7 215 2.875 0.0
7 216 2.875 1.4
7 217 2.875 3.1
7 218 2.875 4.5
7 219 5.625 0.0
7 220 5.625 1.4
7 221 5.625 3.1
7 222 5.625 4.5
7
7 ELEMENT DESCRIPTION:

```

PLATE ELEMENTS ARE NUMBERED
CLOCKWISE AS SHOWN

```

M2 M3
M1 M4

ELEMENT TYPES ARE DESIGNATED AS FOLLOWS:
4 MODE PLAIN ELEMENT TYPE NO. 1
5 MODE LOADED HOLE ELEMENT TYPE NO. 2
4 MODE OPEN HOLE ELEMENT TYPE NO. 3

(NOTE: ENTER M6-8 FOR FOUR MODE ELEMENTS)
ENTER NUMBER OF ELEMENTS IN TOP PLATE
7
7
7 FOR ELEMENT NO 1 ENTER: ELEMENT ID,M1,M2,M3,M4,M5,ELEMENT TYPE
7 101 102 100 100 105 2
7 ENTER ELEMENT THICKNESS
7 0.5
7
7 FOR ELEMENT NO 2 ENTER: ELEMENT ID,M1,M2,M3,M4,M5,ELEMENT TYPE
7 102 102 103 110 100 105 2
7 ENTER ELEMENT THICKNESS
7 0.5
7
7 FOR ELEMENT NO 3 ENTER: ELEMENT ID,M1,M2,M3,M4,M5,ELEMENT TYPE
7 103 103 104 111 110 107 2
7 ENTER ELEMENT THICKNESS
7 0.5
7
7 FOR ELEMENT NO 4 ENTER: ELEMENT ID,M1,M2,M3,M4,M5,ELEMENT TYPE
7 104 100 109 115 115 110 2
7 ENTER ELEMENT THICKNESS
7 0.5
7
7 FOR ELEMENT NO 5 ENTER: ELEMENT ID,M1,M2,M3,M4,M5,ELEMENT TYPE
7 105 100 110 117 116 113 8
7 ENTER ELEMENT THICKNESS
7 0.5
7
7 FOR ELEMENT NO 6 ENTER: ELEMENT ID,M1,M2,M3,M4,M5,ELEMENT TYPE
7 106 110 111 118 117 114 8
7 ENTER ELEMENT THICKNESS
7 0.5

```

Figure 43. SANCJ Input for the Sample Problem in Figure 42 (Continued).

```

FOR ELEMENT NO 7 ENTER: ELEMENT ID,M1,M2,M3,M4,M5,ELEMENT TYPE
107 115 116 120 119 0 1
ENTER ELEMENT THICKNESS
?
0 8
FOR ELEMENT NO 8 ENTER: ELEMENT ID,M1,M2,M3,M4,M5,ELEMENT TYPE
108 115 117 121 120 0 2
ENTER ELEMENT THICKNESS, X AND Y COORDINATES
OF OPEN HOLE AND HOLE RADIUS
?
0.5 4.25 2.25 0.5
FOR ELEMENT NO 9 ENTER: ELEMENT ID,M1,M2,M3,M4,M5,ELEMENT TYPE
109 117 119 122 121 0 1
ENTER ELEMENT THICKNESS
?
0.5
ENTER NUMBER OF ELEMENTS IN BOTTOM PLATE
?
9
FOR ELEMENT NO 10 ENTER: ELEMENT ID,M1,M2,M3,M4,M5,ELEMENT TYPE
201 201 208 209 208 205 2
ENTER ELEMENT LAYOUT NO
?
1
FOR ELEMENT NO 11 ENTER: ELEMENT ID,M1,M2,M3,M4,M5,ELEMENT TYPE
202 202 203 210 209 206 2
ENTER ELEMENT LAYOUT NO
?
1
FOR ELEMENT NO 12 ENTER: ELEMENT ID,M1,M2,M3,M4,M5,ELEMENT TYPE
203 203 204 211 210 207 2
ENTER ELEMENT LAYOUT NO
?
1
FOR ELEMENT NO 13 ENTER: ELEMENT ID,M1,M2,M3,M4,M5,ELEMENT TYPE
204 204 209 216 215 212 2
ENTER ELEMENT LAYOUT NO
?
1
FOR ELEMENT NO 14 ENTER: ELEMENT ID,M1,M2,M3,M4,M5,ELEMENT TYPE
205 209 210 217 219 213 2
ENTER ELEMENT LAYOUT NO
?
1
FOR ELEMENT NO 15 ENTER: ELEMENT ID,M1,M2,M3,M4,M5,ELEMENT TYPE
206 210 211 218 217 214 2
ENTER ELEMENT LAYOUT NO
?
1

```

```

FOR ELEMENT NO 16 ENTER: ELEMENT ID,M1,M2,M3,M4,M5,ELEMENT TYPE
207 215 216 220 219 0 1
ENTER ELEMENT LAYOUT NO
?
1
FOR ELEMENT NO 17 ENTER: ELEMENT ID,M1,M2,M3,M4,M5,ELEMENT TYPE
208 216 217 221 220 0 3
ENTER ELEMENT LAYOUT NUMBER, X AND Y
COORDINATES OF THE OPEN HOLE AND THE HOLE
RADIUS
?
1 4.25 2.25 0.5
FOR ELEMENT NO 18 ENTER: ELEMENT ID,M1,M2,M3,M4,M5,ELEMENT TYPE
209 217 218 222 221 0 1
ENTER ELEMENT LAYOUT NO
?
1
FASTENERS ARE MODELED BY EFFECTIVE
FASTER ELEMENTS WHICH PROVIDE THE
ELASTIC LINK BETWEEN THE TOP AND
BOTTOM PLATES
?
ENTER NUMBER OF FASTENERS IN JOINT
?
6
EFFECTIVE FASTENER ELEMENTS ARE
NUMBERED AS SHOWN: M1 (TOP PLATE)
M2 (BOTTOM PLATE)
WHERE M1 AND M2 CORRESPOND TO THE CENTRAL
HOLES IN LOADED HOLE ELEMENTS
FORMAT: ELEMENT ID, M1, M2
ENTER ELEMENT NO 1
?
101 105 205
ENTER ELEMENT NO 2
?
102 105 205
ENTER ELEMENT NO 3
?
103 107 207
ENTER ELEMENT NO 4
?
104 112 218
ENTER ELEMENT NO 5
?
105 113 213
ENTER ELEMENT NO 6
?
106 114 214

```

Figure 43. SAMCJ Input for the Sample Problem in Figure 42 (Continued).

protruding head) are input.

Twenty-two (22) grid points each are specified in the top and bottom plates (101 to 122 and 201 to 222, respectively), along with their x and y coordinates (see Figure 42). Following this, nine (9) elements are specified in each plate, along with their nodal connectivity and element type information. Nodal connectivity is specified starting from the bottom left node, going clockwise around the element boundary, and ending at the fastener (internal) node. Element 101 in the top plate, for example, has 101, 102, 109 and 108 as its corner nodes, and 105 as its fastener node. The fifth node will be entered as 0 for plain and unloaded hole elements. The element type information follows the fifth node identification. It is 1, 2 and 3 for plain, loaded hole and unloaded hole elements, respectively. The element definitions are succeeded by the definition of six (6) effective fasteners (101 to 106). Fastener 101, for example, is identified as a fastener that connects node 105 in the top plate to node 205 in the bottom plate.

Following the above input, additional element data are specified for the two plates. These include the element thicknesses (for metallic plates) or layup identification number (for laminated plates), for plain and loaded hole elements, with additional information (x and y coordinates of the hole center and the hole radius) for unloaded hole elements. For the sample problem in Figure 42, all the elements in the top plate (metal) are specified to be 0.50 inch thick, and all the elements in the bottom plate (composite) are specified to contain the stacking sequence identified as one (1). Elements 108 and 208 specify the cut-out size and location. The one (1) following this states that groups of identical elements will be specified in the two plates. If two (2) is entered here, all elements will be assumed to be different from one another, resulting in larger computational costs. The entry "1 6 1 1" refers to the number of groups of effective fasteners, loaded hole, unloaded hole and plain elements, respectively, in the top plate. A zero (0) specifies the absence of an element type.

The number of elements in each group, and the corresponding element numbers, are input subsequently. Following this, the number of groups of loaded hole, unloaded hole and plain elements in the bottom plate (6, 1 and 1, respectively) is entered.

The last four lines of input introduce the failure parameters for the materials in the two plates. For metallic plates, the tensile, compressive and shear strengths (250.0D3 each), and the averaging distances for the net section, bearing and shear-out modes of failure (0.5 each) are input. Since the joints were designed to fail the laminated plates, and SAMCJ was developed primarily for the prediction of the strength of bolted laminates, the failure parameters for the metallic plates were input to be arbitrarily high. This information is followed by the failure parameters for the bottom (composite) plate. The first line specifies the fiber directional failure strains for the material under tension (0.012) and compression (0.0175). These values are used by SAMCJ to compute the unnotched laminate tensile, compressive and shear strengths, based on laminated plate theory and the assumption of laminate failure corresponding to the first fiber failure in any of its plies. The last line in Figure 43 specifies the distance over which the longitudinal (0.10 and 0.25) and shear (0.25) stress components are averaged, to predict net section, bearing and shear-out modes of failure, respectively.

3.6 SAMCJ Output Description

For the sample problem introduced in Section 3.5, the SAMCJ code yields the output presented in Figure 44. The initial part of the output reprints critical user-supplied information for verification purposes. Subsequently, SAMCJ prints the x and y components of the element nodal forces for all the elements in the bolted plates. This is followed by a list of the computed joint load levels that correspond to the three failure modes (net section, shear-out and bearing) at every loaded and unloaded hole element location. The smallest among these loads yields the joint failure


```

PROGRAM SANCJ

A SINGLE LAP SHEAR PANEL WILL BE ANALYZED
LOADED IN STATIC TENSION
PLATE NO 1
ALUMINUM
MATERIAL PROPERTIES
E1 -0.1000+00 PSI
E2 -0.1000+00 PSI
G12 -0.1079+00 PSI
NU12 -0.3000+00
NU21 -0.3000+00
PLATE NO 2
A61 3041-C
MATERIAL PROPERTIES
E1 -0.1053+00 PSI
E2 -0.1000+07 PSI
G12 -0.1053+00 PSI
NU12 -0.3000+00
NU21 -0.3000+00
FASTENER DESCRIPTION
STEEL
DIAMETER - 0.3120+00 INCHES
MATERIAL PROPERTIES
E -0.3000+00 PSI
NU -0.3000+00

FAILURE ANALYSIS
AN ALUMINUM SHEAR CRITERION WILL BE USED
PLATE NUMBER 1
METALLIC STRENGTHS
TENSILE STRENGTH - 0.8000+00
COMPRESSION STRENGTH - 0.8000+00
SHEAR STRENGTH - 0.8000+00

CHARACTERISTIC DISTANCES
AGMT - 0.5000+00 INCHES
AGSM - 0.5000+00 INCHES
AGSC - 0.5000+00 INCHES
PLATE NUMBER 8
FIBER STRAIN LIMITS
EPSILON ULT COMP - 0.1750-01
EPSILON ULT TEN - 0.1820-01
GAMMA ULT SHEAR - 0.1200-01

CHARACTERISTIC DISTANCES
AGMT - 0.1000+00 INCHES
AGSM - 0.2500+00 INCHES
AGSC - 0.2500+00 INCHES

PAUSE FOR STIFFNESS MATRIX CALCULATIONS

ELEMENT FORCES
(P APPLIED - 10.000 LBS)

ELEMENT ID 101
GRID FX FY
101 -1.000+00 0.1750-12
102 -1.000+00 -0.2500+00
103 0.0000+00 0.1500+02
104 0.7570+02 0.1250+02
105 0.1053+01 0.1053+00

ELEMENT ID 102
GRID FX FY
102 -1.000+00 0.0240+00
103 -1.000+00 -0.2500+00
104 0.0000+00 -0.5000+01
105 0.0000+00 0.0000+00
106 0.1053+00 0.0000+00

ELEMENT ID 103
GRID FX FY
103 -1.000+00 0.0240+00
104 -1.000+00 -0.2500+00
105 0.0000+00 -0.5000+01
106 0.0000+00 0.0000+00
107 0.1053+00 0.0000+00

```

Figure 44. SANCJ Output for the Problem Defined in Figure 43.

Copy of this document does not
 permit fully legible reproduction

118	0.104D+01	-1.502E+02					
117	0.165D+01	-1.105E+06					
ELEMENT ID 104							
CP10	FX	FV					
109	-7.70E+02	-1.26D+02					
108	-9.18D+02	-1.48D+02					
116	0.245D+01	0.182D+02					
115	0.713D+01	0.747D+01					
112	0.158D+03	0.243D+01					
ELEMENT ID 105							
CR10	FX	FV					
109	-7.76D+02	-5.53D+01					
118	-7.67D+02	0.307D+01					
117	-1.06D+02	0.646D+01					
116	-9.69D+01	-8.15D+01					
113	0.175D+03	0.334D+01					
ELEMENT ID 106							
CP10	FX	FV					
118	-9.75D+02	0.163D+02					
111	-7.58D+02	0.133D+02					
112	0.765D+01	-7.70D+01					
113	0.365D+01	-1.63D+02					
114	0.157D+03	-6.19D+01					
ELEMENT ID 107							
CP10	FX	FV					
115	-7.13D+01	-7.07D+01					
116	0.342D+01	0.347D+01					
120	0.371D+01	0.353D+01					
117	0.702D+12	-7.41D+12					
ELEMENT ID 108							
CR10	FX	FV					
116	0.378D+01	-1.32D+02					
117	0.368D+01	0.133D+02					
121	-3.73D+01	0.357D+01					
120	-3.71D+01	-1.367D+01					
ELEMENT ID 109							
CR10	FX	FV					
117	0.313D+01	-3.08D+01					
118	-7.65D+01	0.719D+01					
122	-1.52D+11	-1.17D+11					
121	0.373D+01	-3.57D+01					
ELEMENT ID 201							
CR10	FX	FV					
210	-8.55D+02	-2.03D+03					
211	-5.48E+02	-8.71D+01					
218	0.154D+03	0.253D+02					
217	0.168E+03	0.573D+01					
214	-1.67D+03	0.619D+01					
ELEMENT ID 202							
CR10	FX	FV					
210	-8.55D+02	-2.03D+03					
211	-5.48E+02	-8.71D+01					
218	0.154D+03	0.253D+02					
217	0.168E+03	0.573D+01					
214	-1.67D+03	0.619D+01					
ELEMENT ID 203							
CR10	FX	FV					
209	-1.11D+03	-2.07D+02					
216	1.16D+03	0.48D+02					
217	2.20D+03	0.453E+02					
216	0.199D+03	-2.44D+02					
213	-1.75D+03	-3.34E+01					
ELEMENT ID 204							
CR10	FX	FV					
208	-5.40D+02	0.919D+01					
209	-8.03D+02	0.247D+02					
215	0.174D+03	-5.89D+01					
215	0.128D+03	-2.46D+02					
212	-1.58E+03	-2.45D+02					
ELEMENT ID 205							
CR10	FX	FV					
209	-1.11D+03	-2.07D+02					
216	1.16D+03	0.48D+02					
217	2.20D+03	0.453E+02					
216	0.199D+03	-2.44D+02					
213	-1.75D+03	-3.34E+01					
ELEMENT ID 206							
CR10	FX	FV					
210	-8.55D+02	-2.03D+03					
211	-5.48E+02	-8.71D+01					
218	0.154D+03	0.253D+02					
217	0.168E+03	0.573D+01					
214	-1.67D+03	0.619D+01					

Figure 44. SANCJ Output for the Problem Defined in Figure 43 (Continued).

ELEMENT ID 207

CRIP	FX	FV
215	1200+03	0.2560+02
216	-1630+03	-1080+01
217	0.1260+03	-1750+02
219	0.1630+03	-1390+01

ELEMENT ID 208

CRIP	FX	FV
216	-2120+03	0.5850+02
217	-2110+03	-5910+02
221	0.2120+03	-1690+02
220	0.2110+03	0.1750+02

ELEMENT ID 209

CRIP	FX	FV
217	-1590+03	0.8410+01
218	-1290+03	-2530+02
222	0.1620+03	-1780+03
221	0.1260+03	0.1690+02

JOINT LOAD LEVELS CORRESPONDING TO NET SECTION (NS), SCAR-OUT (SO) AND BEARING (BR) FAILURES AT EVERY LOADED AND UNLOADED JOINT ELEMENT ARE PREDICTED AS FOLLOWS:

ELEMENT	NS	SO	BR
101	0.5750+06	0.0830+06	0.4590+06
102	0.5460+06	0.7000+06	0.5010+06
103	0.4170+06	0.6260+06	0.4920+06
104	0.2920+07	0.1510+07	0.7300+06
105	0.3110+07	0.1490+07	0.8370+06
106	0.2030+07	0.1710+07	0.7370+06
108	0.1480+08	0.8900+06	0.8270+06
201	0.3400+06	0.1230+06	0.1200+06
202	0.2950+06	0.9700+06	0.1950+06
203	0.5130+06	0.1200+06	0.1320+06
204	0.7040+05	0.6440+05	0.7700+05
205	0.5290+05	0.4110+05	0.5770+05
206	0.8080+05	0.5550+05	0.8130+05
208	0.3730+05	0.1740+06	0.3020+07

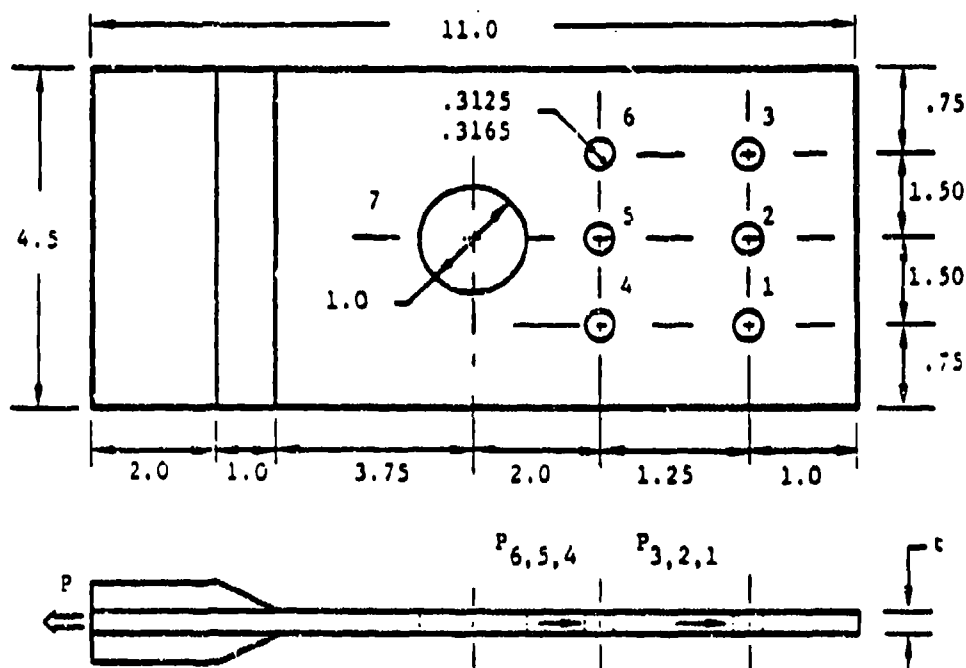
FAILURE IS PREDICTED TO OCCUR IN ELEMENT NUMBER 208 AT AN APPLIED JOINT LOAD VALUE OF 0.3720000+05 LBS

THE PREDICTED FAILURE MODE IS NET SECTION

Figure 44. SAMCJ Output for the Problem Defined in Figure 43 (Concluded).

load, the failure location and the failure mode. For the considered sample problem, a net section failure is predicted across the one inch diameter cut-out (element 208) in the graphite/epoxy plate, at a joint load level of 37.3 kips. Figure 45 compares SAMCJ predictions with test results from Reference 8.

Test Case 243, Static Tension, Single Lap
 40-Ply, 50/40/10 Laminate, $t=0.247$ in., $t_{AL}=0.50$ in.
 $D=5/16$ in., $H_D=1$ in., $S_L/D=S_T/D=4$, $W/D=14.4$, $E/D=3.2$



	SAMCJ PREDICTION	TEST RESULTS (Ref. 2)
P_1/P	0.165	0.162
P_2/P	0.191	0.150
P_3/P	0.165	0.168
P_4/P	0.167	0.177
P_5/P	0.175	0.161
P_6/P	0.168	0.165
$P_{failure}$ (kips)	37.3 (52.7)*	42.0
FAILURE LOCATION	7 (5)	7 and 4, 5, 6
FAILURE MODE(S)	NET SECTION (NET SECTION)	NET SECTION

* Next possible failure mode and location at a higher load level
 Figure 45. Comparison of SAMCJ Predictions for the Sample Problem with
 Test Results from Reference 8.

SECTION 4

DESIGN VERIFICATION OF A BOLTED STRUCTURAL ELEMENT

The design of a highly-loaded structural bolted joint is verified in this section using the analytical tool (SAMCJ computer code) proposed for the recommended design methodology (Section 1.3).

4.1 Description of the Bolted Structural Element

In Reference 5, a bolted joint concept was studied as an alternative to a highly loaded composite-to-titanium, step lap bonded joint. The vertical tail structure of the F/A-18A was used as the baseline for this study. A preliminary design of the bolted structural element, representative of the critical F/A-18A vertical tail root section, was performed based on approximate analyses and available test results. The test element was designed to transfer a design ultimate load of 70.2 kips (obtained from the F/A-18A empennage stress analysis report), and to survive two lifetimes of a representative design spectrum fatigue loading.

The design of the bolted structural element studied in Reference 5 differs from the existing F/A-18A vertical tail root joint significantly. It eliminates the graphite/epoxy skin-to-titanium bonded joint, and directly attaches the skins to the fuselage frame. In doing so, it also uses a light root rib, in contrast to the highly-loaded attachment root rib used in Reference 4. The AS4/3501-6 graphite/epoxy skins of the element have a 41-ply layup away from the attachment location. The skins increase in thickness to a 60-ply layup near the tab region that bolts the vertical tail skin to the fuselage frame. The graphite/epoxy tabs are machined, prior to assembly, to introduce a taper at the joint location. In Reference 5, the fuselage attachment fitting was made out of steel, and the skins were bolted to it using 3/8 inch diameter, countersunk high strength steel bolts. Figure 46 shows a

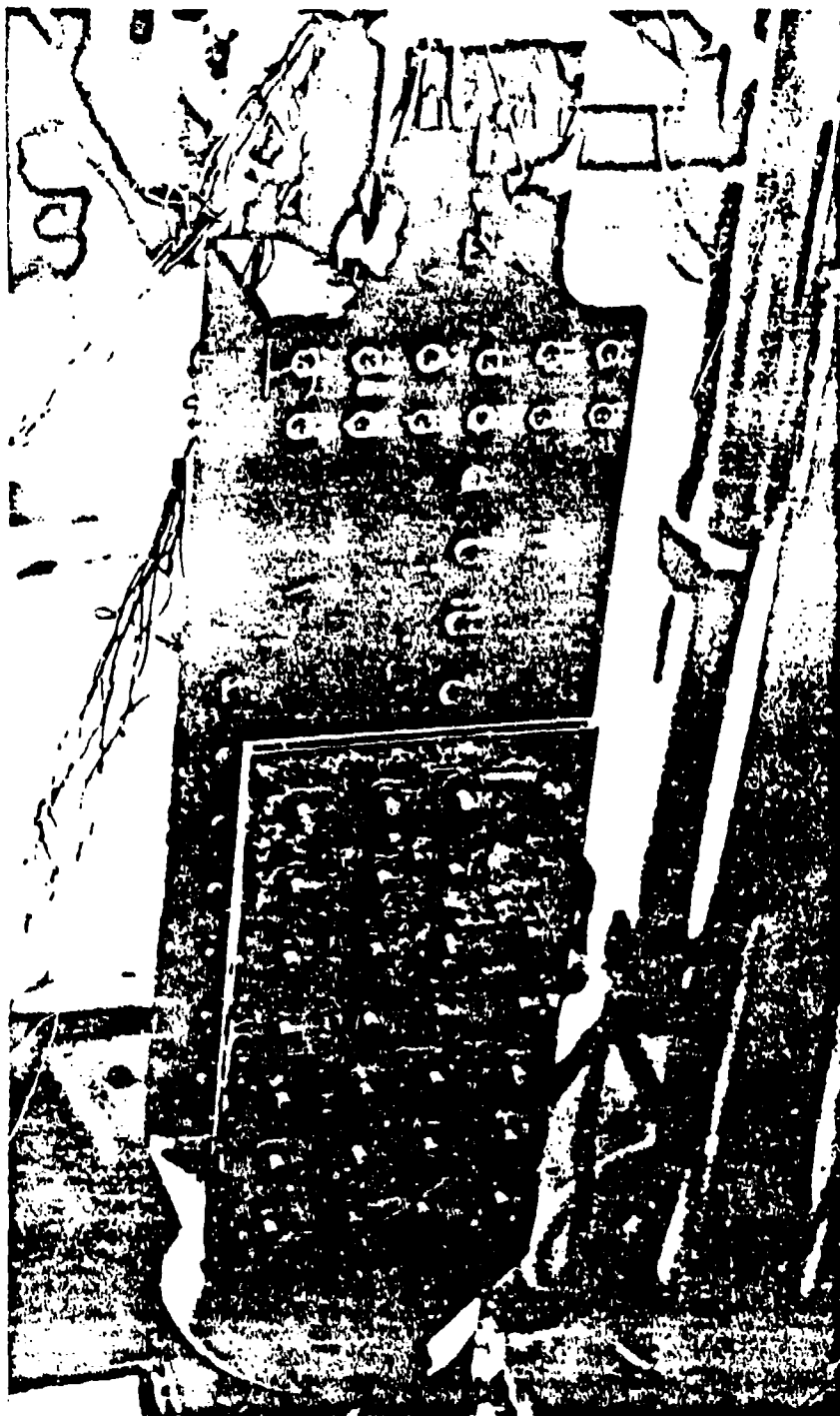


Figure 46. Photograph of an Assembled Test Element.

photograph of an assembled test element. The element spar and the root rib were fabricated using an aluminum alloy.

4.2 Test Results

Elements fabricated based on this preliminary design were subjected to static and fatigue loads in Reference 5. They survived two lifetimes of a spectrum fatigue load that was significantly more severe than the actual F/A-18A vertical tail design spectrum load, and their static strengths were approximately 30% larger than the design ultimate load. During the static test, failure occurred in the graphite/epoxy skin tab in a combined mode (see Figure 47). The observed failure modes were significantly influenced by the tilting or "digging in" of the countersunk fasteners - a phenomenon that cannot be accounted for by the fastener analysis in the SAMCJ computer code.

4.3 Design Verification of the Element Using SAMCJ

The critical vertical tail skin-to-fuselage joint region is analyzed below using the SAMCJ code that is recommended as an analytical design tool. Though the analysis was performed retrospectively, the assumed material and failure parameters are identical to those used in Reference 7.

Figure 48 presents the dimensions of the analyzed graphite/epoxy skin tabs and the fuselage attachment frame. The tapered skin has a $[0_{28}/\pm 45_{12}/90_7]_C$ layup at the top of the tab region. Across the top row of fasteners, it has an average of 58 plies, and across the bottom row of fasteners, it has an average of 52 plies. For analytical purposes, the tapered tab region is modeled as two uniform regions of different thicknesses. The top region is modeled to contain a $[0_{28}/\pm 45_{12}/90_6]_C$ layup, and the bottom region is assumed to be a $[0_{26}/\pm 45_{10}/90_6]_C$ laminate. The average thickness of a ply in the skin was measured to be 0.0049 inch. The fuselage attachment frame is, likewise, divided into a



Figure 47. Photograph of the Tab Region of the Failed Element.

Fuselage
Attachment
Frame

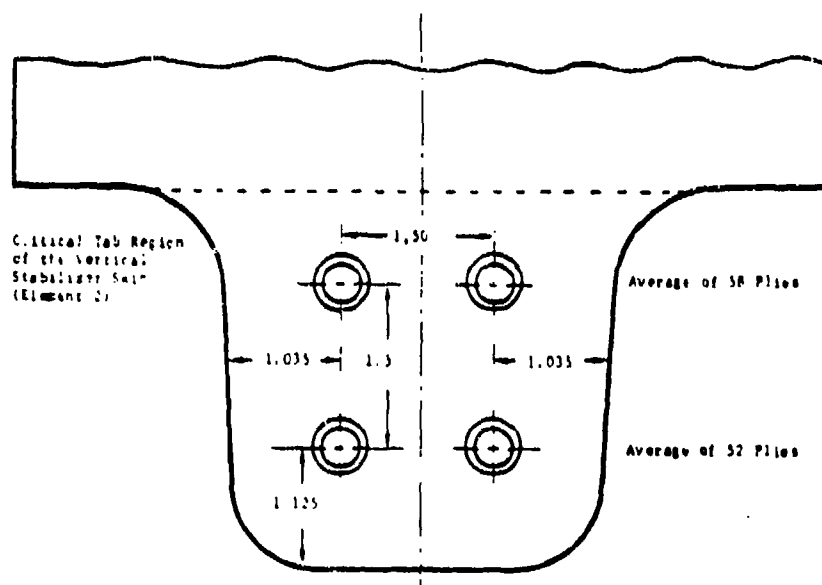
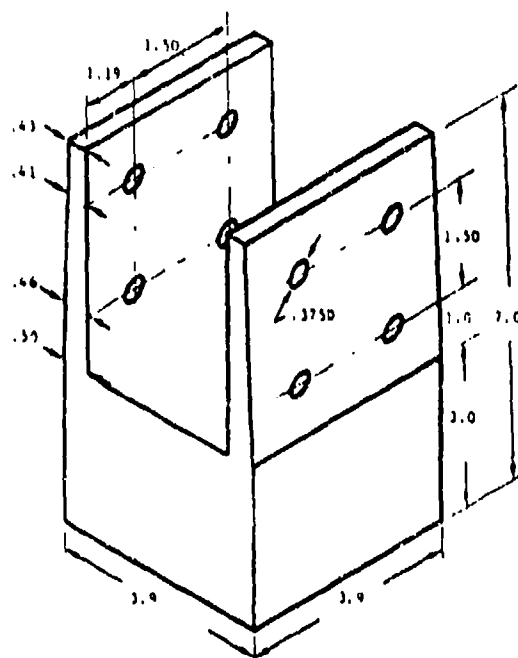


Figure 48. Dimensions of the Critical Skin Tab and the Fuselage Attachment Frame.

0.41 inch thick region and a 0.46 inch thick region (see Figure 48).

The modeled joint segment is half of the symmetric skin tab-to-fuselage attachment. The total joint failure load is, therefore, twice the predicted load. A single shear load transfer between the AS4/3501-6 graphite/epoxy skin tab and the steel attachment frame is analyzed. The graphite/epoxy tab and the steel plate are divided into four elements each. The average width of the slightly tapered tab is used in the analytical model (3.57 in.). The fiber-directional tensile and compressive failure strains for AS4/3501-6 graphite/epoxy are assumed to be 0.012 and 0.0175, respectively (References 7, 13). The characteristic distances for net section, bearing and shear-out failure modes are assumed to be 0.10, 0.25 and 0.25 inch, respectively (Reference 7). The basic AS4/3501-6 lamina properties are assumed to be 18.5 Msi, 1.9 Msi and 0.85 Msi for E_{11} , E_{22} and G_{12} , respectively, and 0.3 for the major Poisson's ratio.

The skins are attached to the fuselage frame by 3/8 inch diameter, countersunk fasteners (100 degree tension head). The fastener analysis in SAMCJ cannot accurately account for the effects of the countersunk head geometry. However, it approximates the actual effects by assuming free rotation at the fastener head location, and requires the user to input an equivalent protruding head fastener diameter. In the discussed element analysis, the average fastener diameter is assumed to be 0.458 inch, to account for the 100 degree tension head geometry.

Analytically predicted load distribution among the fasteners in each tab is presented in Figure 49. The symmetry in the fastener arrangement results in low values for the transverse components of fastener loads (perpendicular to the load direction). Also, the loads in the top row of fasteners are approximately 14% larger than those in the bottom row of fasteners. This leads to a prediction of failure initiation from the top row of fasteners (see Figure 49). The predicted failure site (critical location) is in

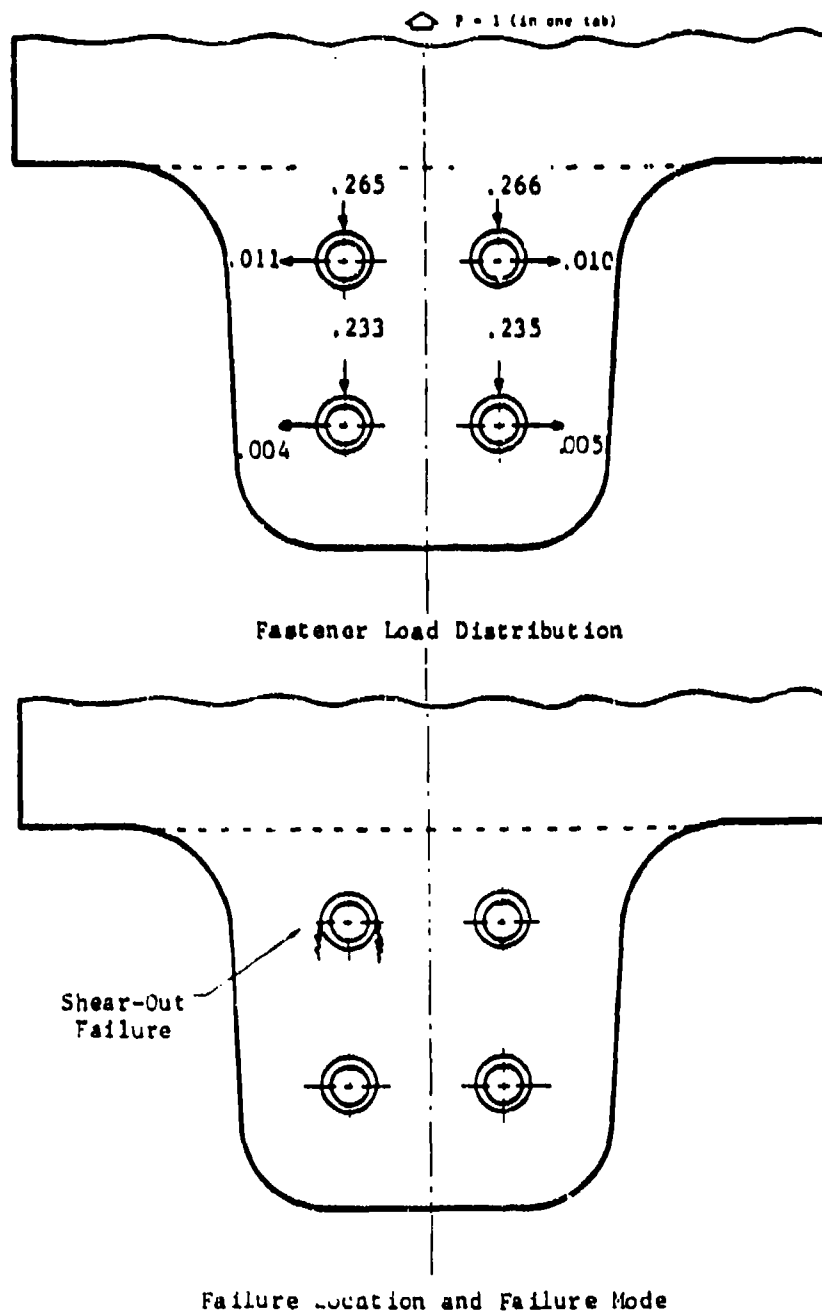


Figure 49. Load Distribution Among Fasteners, Failure Location and Failure Mode in the Graphite/Epoxy Tabs.

agreement with experimental observation.

Figure 50 presents the analytically predicted element load levels to precipitate net section, bearing and shear-out modes of failure at the various fastener locations. The lowest among these provides the element failure load, the failure location and the failure mode. SAMCJ predicts element failure to be caused by a shear-out mode of failure at the top left fastener location in Figure 50. The failure mode observed in Reference 5, however, was severe damage around the fastener hole, introduced by the tilting of the countersunk fasteners (see Figure 47). This included some amount of shear-out and local bearing, and severe delaminations around the fastener hole boundaries. Since SAMCJ cannot account for the severe local three-dimensional stress state introduced by the countersunk fasteners, the predicted failure mode (shear-out) does not correlate well with the observed combined failure mode (partial shear-out, local bearing, and severe delaminations).

Despite the approximate failure mode prediction, however, SAMCJ correctly predicts the failure location, and the failure load predicted by SAMCJ (98.0 kips) is only 7% larger than the measured value (91.8 kips). The approximation of the countersunk fasteners by equivalent protruding head fasteners (larger diameter, unconstrained at the head location), therefore, predicts the element failure load with adequate accuracy. The SAMCJ analysis and the test results in Reference 5 independently verify the 30% margin of safety in the static strength of the test element, due to the approximate analyses used in its preliminary design.

Tab Load = P; Element Load = 2P



The numbers below are the values of the applied element load (2P), in kips to precipitate the three failure modes at each fastener location.

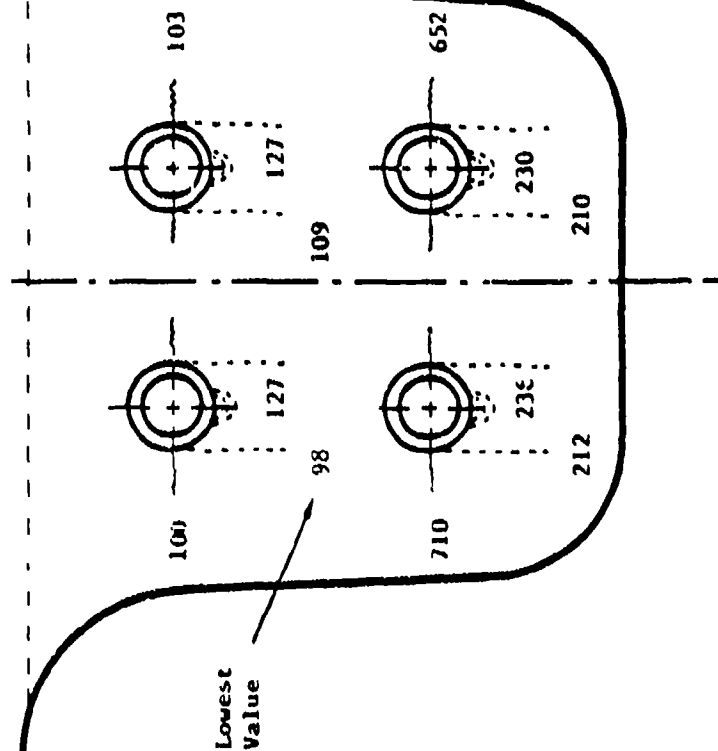


Figure 50. Analytically Predicted Element Load Levels to Precipitate Net Section, Bearing and Shear-Out Modes of Tab Failure at Each Fastener Location.

SECTION 5

CONCLUSIONS

A design guide was developed to enable the user in designing efficient bolted joints in composite structures. The guide highlights general design guidelines for the various parameters that are to be considered in selecting a bolted joint concept. A purely analytical design methodology is presented. It is devoid of complementary test requirements when a previously characterized material is used to fabricate the bolted structure. The design guide also illustrates the use of two computer codes (SASCJ and SAMCJ) that were developed in this Northrop/AFWAL program and are required for design purposes. A listing of these computer codes is appended to this report.

REFERENCES

1. Buchanan, D. L. and Garbo, S. P., "Design of Highly Loaded Composite Joints and Attachment for Wing Structures," NADC-81194-60, August 1981.
2. Watts, D. J., et al., "Development of Composites Technology for Joints and Cutouts in Fuselage Structure of Large Transport Aircraft," Semi-Annual Technical Report Number 1, ACEE-34-PR-3419, October 1984.
3. Kong, S. J., "Conceptual Design of a High Load Transfer Mechanical Attachment for Tail Structures," NADC-81216-60, August 1981.
4. Averill, S. W. and Zamani, H. R., "Development of High Load Joints and Attachments for Composite Structure," NADC-81220-60, August 1983.
5. Ramkumar, R. L., Saether, E. S. and Tossavainen, E., "Design, Fabrication, Testing and Analysis of Bolted Structural Elements," AFWAL-TR-86-XXXX, January 1986.
6. Ramkumar, R. L., et al., "Strength Analysis of Composite and Metallic Plates Bolted Together by a Single Fastener," AFWAL-TR-85-3064, August 1985.
7. Ramkumar, R. L., Saether, E. S., and Appa, K., "Strength Analysis of Laminated and Metallic Plates Bolted Together by Many Fasteners," AFWAL-TR-86-XXXX, January 1986.
8. Ramkumar, R. L. and Tossavainen, E., "Bolted Joints in Composite Structures: Design, Analysis and Verification; Task II Test Results," AFWAL-TR-85-3065, August 1985.

REFERENCES (Concluded)

9. ACEF Composite Structures Technology, Papers by Douglas Aircraft Company, edited by M. Klotzsche, NASA-CR-172359, August 1984.
10. Hart-Smith, L. J., "Mechanically-Fastened Joints for Advanced Composite-Phenomenological Considerations and Simple Analysis," Proceedings of the Fourth Conference on Fibrous Composites in Structural Design, November 1978.
11. Cole, B. and Batah, E. J., "Special Fastener Development for Composite Structure," AFWAL-TR-82-3049, July 1982.
12. Wall, F. T., "Fasteners for Use with Composite Materials," NASC Report Number 78-01, Aerospace Industries Association of America, Inc, January 1978.
13. Garbo, S. P. and Ogonowski, J. M., "Effect of Variances and Manufacturing Tolerances on the Design Strength and Life of Mechanically Fastened Joints," Volume 1, AFWAL-TR-81-3041, April 1981.
14. Ramkumar, R. L. and Tossavainen, E., "Bolted Joints in Composite Structures: Design, Analysis and Verification; Task I Test Results," AFWAL-TR-84-3047, August 1984.
15. Eisenmann, J. R. and Leonhardt, J. L., "Improving Composite Bolted Joint Efficiency by Laminate Tailoring," Joining of Composite Materials, ASTM STP 749, edited by K. T. Kedward, American Society for Testing and Materials, 1981, pp. 117-130.

APPENDIX A

SASCJ Program Listing


```

WRITE(6,6,6)
876 FORMAT(///,10X,' PROGRAM SASCJ',///,
  *' PROGRAM SASCJ PREDICTS FAILURE LOADS OF ',///,
  *' MECHANICALLY FASTENED, COMPOSITE LAMINATE, ',///,
  *' SINGLE OR DOUBLE LAP SHEAR JOINTS. ',///,
  *' PROGRAM ASSUMES THAT INPUT PARAMETERS ARE ',///,
  *' IN ENGLISH UNITS - LENGTHS ARE INPUT ',///,
  *' IN INCHES AND MODULI AND STRENGTHS ARE ',///,
  *' EXPRESSED IN PSI ',///,
  *' )
WRITE(6,401)
401 FORMAT(' ENTER BYPASS RATIO ALPHA: ',///,
  *' ALPHA=0 FOR FULL BEARING ',///,
  *' ALPHA=1 FOR OPEN HOLE ',///,
  *' 0<ALPHA<1 FOR GENERAL BYPASS ')
READ(5,*) BPR
WRITE(6,911)
911 FORMAT(' ENTER: ',///,
  *' 1 FOR STATIC TENSION ',///,
  *' 2 FOR STATIC COMPRESSION ',///,
  *' )
READ(5,*) LTNCM
NLIM=1
IF(BPR.EQ.1.0) GO TO 380
NLIM=2
WRITE(6,400)
400 FORMAT(' ENTER: ',///,
  *' 1 FOR SLS (SINGLE LAP SHEAR)',///,
  *' 2 FOR DLS (DOUBLE LAP SHEAR)',///,
  *' )
READ(5,*) NSDLS
106 FORMAT(A1)
380 CONTINUE
DO 300 K=1,NLIM
  IF(K.EQ.1) WRITE(6,511)
611 FORMAT(' IS THE TOP PLATE A COMPOSITE OR A METAL ? ',///,
  *' ENTER C OR M IN THE FIRST FIELD')
  IF(K.EQ.2) WRITE(6,789)
789 FORMAT(' IS THE BOTTOM PLATE A COMPOSITE OR A METAL ? ',///,
  *' ENTER C OR M IN THE FIRST FIELD')
  READ(5,106) CM(K)
  WRITE(6,203)
203 FORMAT(' INPUT MATERIAL DESCRIPTION OF THIS PLATE ',///,
  *' EX: ASA/3501-6')
  READ(5,204) (MTL(K,1),I=1,15)
204 FORMAT(15A4)
300 CONTINUE
  IF(CM(1).NE.CMC.OR.CM(2).NE.CMC) WRITE(6,721)
721 FORMAT(//,' NOTE: FOR COMPUTATIONAL PURPOSES A ',///,
  *' METALLIC PLATE IS MODELED AS A 30 PLY ',///,
  *' LAMINATE OF 0 DEGREE PLIES WITH ISOTROPIC ',///,
  *' MATERIAL PROPERTIES ',///,
  *' )
  IF(BPR.NE.1.0) WRITE(6,494)
  IF(BPR.EQ.1.0) WRITE(6,495)
494 FORMAT(' NOTE: NUMERICAL DESIGNATIONS FOR THE ',///,
  *' PLATES ARE: ',///,
  *' TOP PLATE = NO 1 ',///,
  *' BOTTOM PLATE = NO 2 ',///,
  *' )
495 FORMAT(' NOTE: A SINGLE PLATE WITH AN OPEN ',///,
  *' HOLE IS DESIGNATED AS PLATE NUMBER 1 ',///,
  *' )
  DO 301 K=1,NLIM
  IF(CM(K).EQ.CMC) GO TO 15
  NPLY(K)=30

```

```

00000560
00000570
00000580
00000590
00000600
00000610
00000620
00000630
00000640
00000650
00000660
00000670
00000680
00000690
00000700
00000710
00000720
00000730
00000740
00000750
00000760
00000770
00000780
00000790
00000800
00000810
00000820
00000830
00000840
00000850
00000860
00000870
00000880
00000890
00000900
00000910
00000920
00000930
00000940
00000950
00000960
00000970
00000980
00000990
00001000
00001010
00001020
00001030
00001040
00001050
00001060
00001070
00001080
00001090
00001100
00001110
00001120
00001130
00001140
00001150

```

GO TO 301	00001160
15 CONTINUE	00001170
IF(NSDLS.EQ.2.AND.K.EQ.1) WRITE(6,932)	00001180
932 FORMAT(/,' NOTE: FOR THE DOUBLE LAP SHEAR CASE HAVING ',/,	00001190
*' A COMPOSITE PLATE NUMBER 2, ENTER ONLY HALF',/,	00001200
*' OF THE LAYUP - IE HALF THE NUMBER OF ACTUAL',/,	00001210
*' PLIES ')	00001220
WRITE(6,205) K	00001230
205 FORMAT(' INPUT NUMBER OF PLIES IN PLATE NO',I5,/,	00001240
*' (N > OR = 2)')	00001250
READ(5,*) NPLY(K)	00001260
NLIM2=2*NPLY(K)+1	00001270
301 CONTINUE	00001280
DO 302 K=1,NLIM	00001290
IF(CM(K).EQ.CMC) GO TO 25	00001300
IF(NSDLS.EQ.2.AND.K.EQ.2) WRITE(6,933)	00001310
933 FORMAT(/,' FOR THE DOUBLE LAP SHEAR CASE HAVING',/,	00001320
*' A METALLIC PLATE NUMBER TWO, ENTER HALF THE ',/,	00001330
*' ACTUAL PLATE THICKNESS ')	00001340
WRITE(6,35) K	00001350
35 FORMAT(' INPUT THICKNESS OF PLATE NO',I5)	00001360
READ(5,*) A1	00001370
H(K)=A1/NPLY(K)	00001380
GO TO 302	00001390
25 CONTINUE	00001400
WRITE(6,260) K	00001410
260 FORMAT(' INPUT PLY THICKNESS IN PLATE NO',I5)	00001420
READ(5,*) H(K)	00001430
302 CONTINUE	00001440
DO 303 K=1,NLIM	00001450
IF(CM(K).EQ.CMC) GO TO 45	00001460
NUMPLY(K)=1	00001470
GO TO 303	00001480
45 CONTINUE	00001490
WRITE(6,207) K	00001500
207 FORMAT(' INPUT NUMBER OF DISTINCT PLY ORIENTATIONS',/,	00001510
*' IN PLATE NO',I5,	00001520
READ(5,*) NUMPLY(K)	00001530
303 CONTINUE	00001540
DO 209 K=1,NLIM	00001550
IF(CM(K).EQ.CMC) GO TO 55	00001560
ANG(1,K)=0.	00001570
GO TO 209	00001580
55 CONTINUE	00001590
WRITE(6,487) K	00001600
487 FORMAT(/,' FOR PLATE NUMBER',I5,' ',/,	00001610
N=NUMPLY(K)	00001620
DO 209 L=1,N	00001630
WRITE(6,206) L	00001640
206 FORMAT(' INPUT ORIENTATION OF PLY TYPE NO',I5)	00001650
READ(5,*) ANG(L,K)	00001660
209 CONTINUE	00001670
DO 305 K=1,NLIM	00001680
IF(CM(K).EQ.CMC) GO TO 65	00001690
NN=NPLY(K)	00001700
DO 75 IJ=1,NN	00001710
75 IPLY(IJ,K)=1	00001720
GO TO 305	00001730
65 CONTINUE	00001740
WRITE(6,210) K	00001750

```

210 FORMAT(' INPUT TYPE OF PLY IN PLATE NO',I5,' FROM TOP',/,
* TO BOTTOM
*5X,'PLY TYPE',10X,'ORIENTATION')
N=NUMPLY(K)
DO 212 L=1,N
WRITE(6,713) L,ANG(L,K)
213 FORMAT(5X,I5,10X,F7.2,' DEGREES')
212 CONTINUE
WRITE(6,711)
711 FORMAT(/)
N=NPPLY(K)
DO 215 I=1,N
WRITE(6,214) I
214 FORMAT(' INPUT TYPE OF PLY FOR PLY NO',I5)
READ(5,*) IPPLY(I,K)
215 CONTINUE
305 CONTINUE
DO 306 K=1,NLIM
WRITE(6,216) K
216 FORMAT(' INPUT THE ENGINEERING PROPERTIES OF PLATE NO',I5)
IF(CMC(K).EQ.CMC) GO TO 35
WRITE(6,95)
95 FORMAT(' INPUT YOUNGS MODULUS AND POISSONS RATIO')
READ(5,*) E1(K),V12(K)
E2(K)=E1(K)
V21(K)=V12(K)*E2(K)/E1(K)
G12(K)=E1(K)/(2.*(1.+V12(K)))
GO TO 306
85 CONTINUE
WRITE(6,217)
217 FORMAT(' INPUT YOUNGS MODULI, E1 AND E2')
READ(5,*) E1(K),E2(K)
WRITE(6,218)
218 FORMAT(' INPUT THE SHEAR MODULUS AND MAJOR POISSONS RATIO')
READ(5,*) G12(K),V12(K)
V21(K)=V12(K)*E2(K)/E1(K)
306 CONTINUE
IF(UPR.NE.1.0) GO TO 930
WRITE(6,844)
844 FORMAT(' INPUT HOLE DIAMETER')
READ(5,*) FASD
GO TO 360
930 CONTINUE
WRITE(6,250)
250 FORMAT(' INPUT MATERIAL DESCRIPTION FOR FASTENER')
READ(5,251) (MTL(J,I),I=1,15)
251 FORMAT(15A4)
WRITE(6,252)
252 FORMAT(' INPUT YOUNG MODULUS AND POISSONS RATIO FOR',/,
* THE FASTENER')
READ(5,*) FASE,FASV
WRITE(6,253)
253 FORMAT(' INPUT THE DIAMETER OF THE FASTENER')
READ(5,*) FASD
WRITE(6,888)
888 FORMAT(/,' FASTENER TYPE ',/,
* ENTER: 1 FOR PROTRUDING HEAD ',/,
* 2 FOR COUNTERSUNK HEAD ')
READ(5,*) HFTYP
R(1)=1.0D10

```

```

00001760
00001770
00001780
00001790
00001800
00001810
00001820
00001830
00001840
00001850
00001860
00001870
00001880
00001890
00001900
00001910
00001920
00001930
00001940
00001950
00001960
00001970
00001980
00001990
0002000
0002010
0002020
0002030
0002040
0002050
0002060
0002070
0002080
0002090
0002100
0002110
0002120
0002130
0002140
0002150
0002160
0002170
0002180
0002190
0002200
0002210
0002220
0002230
0002240
0002250
0002260
0002270
0002280
0002290
0002300
0002310
0002320
0002330
0002340
0002350

```

```

R(2)=1.0E-10
IF(NFTYP.EQ.1) GO TO 360
WRITE(6,889)
889 FORMAT(/,' ENTER PLATE WHICH CONTAINS THE COUNTERSUNK',/,
* HEAD (OPPOSITE PLATE ASSUMES THE NUT HEAD) ',/,
* ENTER: 1 FOR TOP PLATE ',/,
* 2 FOR BOTTOM PLATE ')
READ(5,M) N
R(N)=0.0D0
360 CONTINUE

00002360
00002370
00002380
00002390
00002400
00002410
00002420
00002430
00002440
00002450
00002460
00002470
00002480
00002490
00002500
00002510
00002520
00002530
00002540
00002550
00002560
00002570
00002580
00002590
00002600
00002610
00002620
00002630
00002640
00002650
00002660
00002670
00002680
00002690
00002700
00002710
00002720
00002730
00002740
00002750
00002760
00002770
00002780
00002790
00002800
00002810
00002820
00002830
00002840
00002850
00002860
00002870
00002880
00002890
00002900
00002910
00002920
00002930
00002940
00002950

0000
READ IN GEOMETRY AND BOUNDARY DATA

AX=FASD/2.0D0
BX=AX
WRITE(6,856)
856 FORMAT(' PLATE GEOMETRIES ARE SPECIFIED BY ',/,
* INPUTTING THE COORDINATES OF THE CORNER',/,
* VERTICES. NOTE: THE ORIGIN IS AT THE FASTENER',/,
* CENTER ; INPUT COORDINATES ACCORDINGLY',/,
*
* V3 HOLE V2
* CENTROID V1
* V4
* APPLIED LOAD CONVENTION:
* FOR PLATE NO 1 (TOP) NORMAL LOADS ARE APPLIED ',/,
* BETWEEN V3 AND V4
* FOR PLATE NO 2 (BOTTOM) NORMAL LOADS ARE APPLIED ',/,
* BETWEEN V1 AND V2
DO 480 K=1,NLIM
WRITE(6,734) K
734 FORMAT(' FOR PLATE NUMBER ',I5,' ',/,)
DO 110 I=1,4
WRITE(6,290) I
290 FORMAT(' ENTER X,Y COORDINATES OF V',I4)
READ(5,M) XC(K,I),YC(K,I)
110 CONTINUE
IF(K.EQ.2) GO TO 841
A1=XC(1,1)
B1=YC(1,1)
A2=XC(1,2)
B2=YC(1,2)
XC(1,1)=XC(1,4)
YC(1,1)=YC(1,4)
XC(1,2)=XC(1,3)
YC(1,2)=YC(1,3)
XC(1,4)=A1
YC(1,4)=B1
XC(1,3)=A2
YC(1,3)=B2
841 CONTINUE
WTH=YC(K,2)-YC(K,1)
480 CONTINUE
IF(BPR.EQ.0.0.OR.BPR.EQ.1.0) GO TO 567
WRITE(6,741)
741 FORMAT(' SELECT FAILURE CRITERION: ',/,
* ENTER 1 FOR POINT STRESS CRITERION ',/,
* ENTER 2 FOR AVERAGE STRESS CRITERION ')
READ(5,M) NPT

```

```

IF(NPT.EQ.1) NOPT4=2
IF(NPT.EQ.2) NOPT4=4
GO TO 601
567 CONTINUE
WRITE(6,220)
220 FORMAT(' SELECT FAILURE CRITERION ',//,
* ENTER 1 FOR HOFFMAN/ TSAI-HILL CRITERION ',//,
* ENTER 2 FOR POINT STRESS CRITERION ',//,
* ENTER 3 FOR MAXIMUM STRAIN CRITERION ',//,
* ENTER 4 FOR AVERAGE STRESS CRITERION')
READ(5,*) NOPT4
601 CONTINUE
IF(NOPT4.EQ.2.OR.NOPT4.EQ.4) GO TO 221
DO 412 K=1,NLIM
WRITE(6,222) K
222 FORMAT(' FOR PLATE NUMBER ',I5,' ENTER RADIUS OF ',//,
* CHARACTERISTIC CIRCLE AT WHICH STRESSES ARE',//,
* TO BE COMPUTED TO PREDICT FAILURE')
READ(5,*) RCA(K)
RCD(K)=RCA(K)
HRCOUT(K)=50
IF(NOPT4.EQ.3) GO TO 591
WRITE(6,834)
834 FORMAT(' ENTER THE FAILURE INDEXES FOR THE ',//,
* HOFFMAN/ TSAI-HILL CRITERIA ',//,
* NOTE: FOR USING TSAI-HILL SET EQUAL THE COMPRESSION ',//,
* AND TENSION ULTIMATES IN SIGMA X AND ',//,
* ENTER: SIGMA X ULTIMATE-COMPRESSION ',//,
* SIGMA X ULTIMATE-TENSION ',//,
* SIGMA Y ULTIMATE-COMPRESSION ',//,
* SIGMA Y ULTIMATE-TENSION ',//,
* SIGMA XY ULTIMATE ')
READ(5,*)(HFMCI,K),I=1,5)
GO TO 412
591 CONTINUE
WRITE(6,393) K
393 FORMAT(' ENTER MAXIMUM STRAIN ALLOWABLE FOR',//,
* PLATE NUMBER ',I7,' (UNITS: IN/IN)')
READ(5,*) SALON(K)
412 CONTINUE
IF(NOPT4.EQ.3) GO TO 391
IF(NOPT4.EQ.1) GO TO 391
GO TO 262
221 CONTINUE
IF(NOPT4.EQ.2) WRITE(6,555)
IF(NOPT4.EQ.4) WRITE(6,556)
555 FORMAT(/,' POINT STRESS CRITERION ',//)
556 FORMAT(/,' AVERAGE STRESS CRITERION ',//,
* AO IS THE CHARACTERISTIC DISTANCE OVER WHICH',//,
* STRESSES ARE AVERAGED AND COMPARED WITH UNNOTCHED',//,
* STRENGTHS TO PREDICT FAILURE')
DO 226 K=1,NLIM
IF(BPR.NE.0.0.AND.BPR.NE.1.0) GO TO 531
WRITE(6,225) K
225 FORMAT(' INPUT AO FOR EACH OF THE THREE PLY FAILURE',//,
* MODES OF PLATE NO',I5,/,
* AONT = NET SECTION ',//,
* AOBK = BEARING ',//,
* AOSO = SHEAR OUT ',//)
N=NUMPLY(K)

```

```

00002960
00002970
00002980
00002990
00003000
00003010
00003020
00003030
00003040
00003050
00003060
00003070
00003080
00003090
00003100
00003110
00003120
00003130
00003140
00003150
00003160
00003170
00003180
00003190
00003200
00003210
00003220
00003230
00003240
00003250
00003260
00003270
00003280
00003290
00003300
00003310
00003320
00003330
00003340
00003350
00003360
00003370
00003380
00003390
00003400
00003410
00003420
00003430
00003440
00003450
00003460
00003470
00003480
00003490
00003500
00003510
00003520
00003530
00003540
00003550

```



```

WRITE(6,227)
227 FORMAT(' INPUT AONT,AOBR,AND AOSO')
READ(5,M) DONT(K),DOBR(K),DOSQ(K)
GO TO 226
531 CONTINUE
WRITE(6,532) K
532 FORMAT(' ENTER AO VALUES CORRESPONDING TO THE THREE',/,
* PLY FAILURE MODES IN PLATE NO ',I2,/,
* AONT = NET SECTION
* AOBR = BEARING
* AOSO = SHEAR OUT ')
WRITE(6,533)
533 FORMAT(' INPUT AONT, AOBR, AOSO ')
READ(5,M) AONT(1,K),AOBR(1,K),AOSO(1,K)
AONT(2,K)=AONT(1,K)
AOBR(2,K)=AOBR(1,K)
AOSO(2,K)=AOSO(1,K)
IF(K.EQ.1) WRITE(6,554)
554 FORMAT(/, ' TO AVOID LENGTHY RUN TIMES DUE TO ',/,
* STRESS FIELD RECOMPUTATION SPECIFY THE ',/,
* NUMBER OF ULTIMATE PLY FAILURES AFTER ',/,
* WHICH JOINT FAILURE WILL BE PREDICTED ',/,
* ENTER: NO OF ULTIMATE FAILURES ')
IF(K.EQ.1) READ(5,M) NULTF
226 CONTINUE
291 CONTINUE
NOPT1=1
IF(BPR.EQ.1.0) NOPT1=2
DO 229 K=1,NLIM
IF(BPR.NE.1.0.OR.(BPR.EQ.1.0.AND.NOPT1.EQ.2)) GO TO 670
GO TO 671
670 CONTINUE
IF(CM(K).EQ.CMC) GO TO 672
WRITE(6,228) K
228 FORMAT(' FOR PLATE NUMBER ',I5,' ENTER THE THREE STRENGTHS ',/,
* REQUIRED TO PREDICT THE THREE FAILURE MODES ',/,
* FNST=UNNOTCHED STRENGTH IN TENSION ',/,
* FNSC=UNNOTCHED STRENGTH IN COMPRESSION',/,
* FSO=UNNOTCHED STRENGTH IN SHEAR-OUT',/,
* INPUT FNST,FNSC,FSO ')
READ(5,M) AF1,AF2,AF4
GO TO 673
672 WRITE(6,674) K
674 FORMAT(' FOR PLATE NO ',I5,' ENTER FIBER ULTIMATE',/,
* STRAIN VALUES
* EPSILON ULT IN COMPRESSION
* EPSILON ULT IN TENSION
* GAMMA ULT IN SHEAR
* (UNITS: IN/IN)')
READ(5,M) ES1(K),ES2(K),ESS(K)
CALL STRTH(H,ES1,ES2,ESS,AF1,AF2,AF4,K)
675 CONTINUE
AF3=AF2
NP=NUMPLY(K)
DO 666 IL=1,NP
PSTC(1,IL,K)=AF1
PSTC(2,IL,K)=AF2
PSTC(3,IL,K)=AF3
PSTC(4,IL,K)=AF4
666 CONTINUE

```

```

00003560
00003570
00003580
00003590
00003600
00003610
00003620
00003630
00003640
00003650
00003660
00003670
00003680
00003690
00003700
00003710
00003720
00003730
00003740
00003750
00003760
00003770
00003780
00003790
00003800
00003810
00003820
00003830
00003840
00003850
00003860
00003870
00003880
00003890
00003900
00003910
00003920
00003930
00003940
00003950
00003960
00003970
00003980
00003990
00004000
00004010
00004020
00004030
00004040
00004050
00004060
00004070
00004080
00004090
00004100
00004110
00004120
00004130
00004140
00004150

```

229	CONTINUE	00004160
671	CONTINUE	00004170
	IF(NOPT4.NE.4) GO TO 261	00004180
C		00004190
C	NUMBER OF DIVISIONS FOR STRESS AVERAGING	00004200
C	IS SET EQUAL TO 50	00004210
	NAVD=50	00004220
261	CONTINUE	00004230
	IF(BPR.EQ.1.0) GO TO 262	00004240
	DO 319 K=1,NLIM	00004250
	N=NUMPLY(K)	00004260
	WRITE(6,320) K	00004270
320	FORMAT(' SASCJ ASSUMES A BILINEAR PLY BEHAVIOR. THE ',,,	00004280
	* INITIAL MODULUS, K1, IS COMPUTED BY THE CODE. ',,,	00004290
	* THE REDUCED MODULUS, K2, FOR INITIAL FAILURE ',,,	00004300
	* IN NET SECTION, SHEAROUT OR BEARING IS COMPUTED ',,,	00004310
	* BY THE FORMULA $K2=ALPHAK1$. ',,,	00004320
	* FOR PLATE NUMBER '15,' INPUT ALPHA VALUES FOR ',,,	00004330
	* NET SECTION, SHEAROUT AND BEARING FAILURE ')	00004340
	READ(5,*) AF1,AF2,AF3	00004350
	DO 321 I=1,N	00004360
	DELNS(I,K)=AF1	00004370
	DELSR(I,K)=AF2	00004380
	DELSO(I,K)=AF3	00004390
321	CONTINUE	00004400
	WRITE(6,339)	00004410
339	FORMAT(' INPUT SCALE FACTORS FOR P ULTIMATE ',,,	00004420
	* CALCULATION SUCH THAT $P(ULT)=BETA*P(INITIAL)$ ',,,	00004430
	* INPUT BETA1 FOR NET SECTION ULTIMATE ',,,	00004440
	* BETA2 FOR BEARING ULTIMATE ',,,	00004450
	* BETA3 FOR SHEAROUT ULTIMATE ')	00004460
	READ(5,*) PALT(3,K),PALT(2,K),PALT(1,K)	00004470
319	CONTINUE	00004480
391	CONTINUE	00004490
	IF(BPR.NE.0.0) GO TO 262	00004500
	DO 312 K=1,NLIM	00004510
	GAMD(K)=10.0	00004520
	IF(CMK(K).NE.CMC) GO TO 312	00004530
	WRITE(6,231) K	00004540
231	FORMAT(' INPUT THE APPROXIMATE INTERLAMINAR SHEAR STRAIN ',,,	00004550
	* ULTIMATE FOR DELAMINATION PREDICTION IN PLATE NO '15, ',,,	00004560
	* (UNITS: IN/IN) ')	00004570
	READ(5,*) GAMD(K)	00004580
312	CONTINUE	00004590
262	CONTINUE	00004600
C		00004610
C	CASE HEADING	00004620
C		00004630
	WRITE(6,143)	00004640
143	FORMAT('PROGRAM SASCJ',,,)	00004650
	IF(NSDLS.EQ.1.AND.BPR.NE.1.0) WRITE(6,633)	00004660
	IF(NSDLS.EQ.2.AND.BPR.NE.1.0) WRITE(6,634)	00004670
633	FORMAT(2X,'A SINGLE LAP SHEAR JOINT WILL BE ANALYZED',,,)	00004680
634	FORMAT(2X,'A DOUBLE LAP SHEAR JOINT WILL BE ANALYZED',,,)	00004690
	IF(BPR.EQ.0.0) WRITE(6,881)	00004700
	IF(BPR.EQ.1.0) WRITE(6,882)	00004710
	IF(BPR.NE.0.0.AND.BPR.NE.1.0) WRITE(6,883) BPR	00004720
881	FORMAT(2X,'WITH A LOADED HOLE',,,)	00004730
882	FORMAT(2X,'WITH AN OPEN HOLE',,,)	00004740
		00004750

883	FORMAT(2X,'WITH A PARTIALLY LOADED HOLE',//	00004760
	*2X,'BYPASS RATIO = ',D9.3,//)	00004770
	IF(LTNCM.EQ.1) WRITE(6,823)	00004780
	IF(LTNCM.EQ.2) WRITE(6,824)	00004790
823	FORMAT(2X,'LOADED IN STATIC TENSION',//)	00004800
824	FORMAT(2X,'LOADED IN STATIC COMPRESSION',//)	00004810
	DO 605 I=1,NLIM	00004820
	WRITE(6,600) I	00004830
600	FORMAT(10X,'PLATE NO ',I5,' ',//)	00004840
	WRITE(6,689) (MTL(I,J),J=1,15)	00004850
689	FORMAT(2X,15A4,//)	00004860
	HT=NPLY(I)*H(I)	00004870
	WRITE(6,602) HT	00004880
602	FORMAT(2X,'T = ',D9.3,' INCHES',//)	00004890
	WRITE(6,603) E1(I),E2(I),G12(I),V12(I),V21(I)	00004900
603	FORMAT(2X,'MATERIAL PROPERTIES',//,//)	00004910
	*10X,'E1 = ',D9.3,' PSI',//	00004920
	*10X,'E2 = ',D9.3,' PSI',//	00004930
	*10X,'G12 = ',D9.3,' PSI',//	00004940
	*10X,'NU12 = ',D9.3,//	00004950
	*10X,'NU21 = ',D9.3,//	00004960
605	CONTINUE	00004970
	IF(DPR.EQ.1.0) GO TO 708	00004980
	WRITE(6,606)	00004990
606	FORMAT(10X,'FASTENER',//)	00005000
	WRITE(6,607) (MTL(3,J),J=1,15)	00005010
607	FORMAT(2X,15A4,//)	00005020
	WRITE(6,608) FASD	00005030
608	FORMAT(2X,' DIAMETER = ',D9.3,' INCHES',//)	00005040
	WRITE(6,609) FASE,FASV	00005050
609	FORMAT(2X,' MATERIAL PROPERTIES',//,//)	00005060
	*10X,'E = ',D9.3,' PSI',//	00005070
	*10X,'MU = ',D9.3,//	00005080
708	CONTINUE	00005090
	WRITE(6,923)	00005100
923	FORMAT(//,10X,'FAILURE ANALYSIS',//)	00005110
	IF(NOPT4.EQ.2.OR.NOPT4.EQ.4) GO TO 621	00005120
	IF(NOPT4.EQ.3) GO TO 821	00005130
	WRITE(6,622)	00005140
622	FORMAT(2X,'THE HOFFMAN-TSAI-HILL CRITERION WILL BE USED',//)	00005150
	DO 623 J=1,NLIM	00005160
	WRITE(6,624) J,RCA(J)	00005170
624	FORMAT(2X,'PLATE NUMBER ',I5,//,	00005180
	*2X,'CHARACTERISTIC RADIUS = ',D9.3,' INCHES')	00005190
	WRITE(6,790)	00005200
790	FORMAT(//,16X,' ULTIMATE STRESSES',	00005210
	*10X,'TENSION',18X,'COMPRESSION')	00005220
	WRITE(6,625) (HFMC(I,J), I=1,5),HFMC(5,J)	00005230
625	FORMAT(//,2X,'SIGMA X = ',D9.3,' PSI',5X,'SIGMA X = ',	00005240
	*D9.3,' PSI',//,	00005250
	*2X,'SIGMA Y = ',D9.3,' PSI',5X,'SIGMA Y = ',	00005260
	*D9.3,' PSI',//,	00005270
	*2X,'SIGMA S = ',D9.3,' PSI',5X,'SIGMA S = ',	00005280
	*D9.3,' PSI',//)	00005290
623	CONTINUE	00005300
	GO TO 627	00005310
621	CONTINUE	00005320
	IF(NOPT4.EQ.2) WRITE(6,628)	00005330
628	FORMAT(2X,'A POINT STRESS CRITERION WILL BE USED',//)	00005340
	IF(NOPT4.EQ.4) WRITE(6,558)	00005350

558	FORMAT(2X, 'AN AVERAGE STRESS CRITERION WILL BE USED',//)	00005360
	DO 631 I=1,NLIM	00005370
	WRITE(6,632) I	00005380
632	FORMAT(2X, 'PLATE NUMBER', I5,//)	00005390
	NP=NUMPLY(I)	00005400
	WRITE(6,713)	00005410
713	FORMAT(//, 2X, 'LAMINATE STRENGTH',//)	00005420
776	WRITE(6,677) (PSTC(LL,1,I), LL=1,4)	00005430
677	FORMAT(2X, 'NET SECTION ULYIMATE (TEN) =', D9.3, ' PSI',//)	00005440
	*' NET SECTION ULYIMATE (COMP) =', D9.3, ' PSI',//	00005450
	*2X, 'BEARING ULYIMATE =', D9.3, ' PSI',//	00005460
	*2X, 'SHEAROUT ULYIMATE =', D9.3, ' PSI',//	00005470
	IF(BPR.NE.0.0.AND.BPR.NE.1.0) GO TO 561	00005480
	WRITE(6,644)	00005490
644	FORMAT(2X, 'CHARACTERISTIC DISTANCES',//)	00005500
	WRITE(6,645) DONT(I), DOBR(I), DOSO(I)	00005510
645	FORMAT(2X, ' DONT =', D9.3, ' INCHES',//)	00005520
	*2X, ' DOBR =', D9.3, ' INCHES',//	00005530
	*2X, ' DOSO =', D9.3, ' INCHES',//	00005540
	GO TO 631	00005550
561	WRITE(6,562)	00005560
562	FORMAT(2X, 'CHARACTERISTIC DISTANCES',//)	00005570
	WRITE(6,564) AONT(1,1), AOBX(1,1), AOSO(1,1)	00005580
564	FORMAT(2X, ' AONT =', D9.3, ' INCHES',//)	00005590
	*' AOBX =', D9.3, ' INCHES',//	00005600
	*' AOSO =', D9.3, ' INCHES',//	00005610
631	CONTINUE	00005620
	GO TO 627	00005630
821	CONTINUE	00005640
	WRITE(6,822)	00005650
822	FORMAT(2X, 'MAXIMUM STRAIN CRITERION WILL BE USED',//)	00005660
	DO 887 II=1,NLIM	00005670
	WRITE(6,858) II, RCA(II)	00005680
858	FORMAT(2X, 'PLATE NUMBER', I5,//)	00005690
	*2X, 'CHARACTERISTIC RADIUS =', D9.3, ' INCHES',//	00005700
	WRITE(6,825) SALOW(II)	00005710
825	FORMAT(//, 8X, 'STRAIN ULYIMATE =', D9.3, ' IN/IN',//)	00005720
887	CONTINUE	00005730
627	CONTINUE	00005740
		00005750
	CALCULATE THE PLY FOUNDATION MODULI AND	00005760
	FAILURE LOADS	00005770
		00005780
	NBP=1	00005790
	IF(BPR.NE.0.0.AND.BPR.NE.1.0) NBP=2	00005800
	IF(NBP.EQ.1) NLIM2=1	00005810
	DO 71 LOM=1,NLIM2	00005820
	DO 22 IL=1,NBP	00005830
	DO 20 K=1,NLIM	00005840
		00005850
	INITIALIZE PARAMETERS FOR COLLOCATION	00005860
		00005870
	NT=7	00005880
	NOUT=50	00005890
	NCOI=10	00005900
	NB=NOUT+4*NCOI	00005910
		00005920
	CONTINUE CASE HEADING	00005930
		00005940
	IF(LOM.GT.1) GO TO 23	00005950

	IF(IL.EQ.1) GO TO 23	00005960
	WRITE(6,871) K	00005970
871	FORMAT(1,5X,' GEOMETRY OF PLATE NO ',I5.1,' /')	00005980
	WRITE(6,872)	00005990
872	FORMAT(' COORDINATES OF CORNER VERTEXES ',/)	00006000
	IF(K.EQ.1) WRITE(6,873) XC(K,2),YC(K,2),XC(K,3),YC(K,3)	00006010
	IF(K.EQ.1) WRITE(6,874) XC(K,1),YC(K,1),XC(K,4),YC(K,4)	00006020
	IF(K.EQ.2) WRITE(6,873) XC(K,3),YC(K,3),XC(K,2),YC(K,2)	00006030
	IF(K.EQ.2) WRITE(6,874) XC(K,4),YC(K,4),XC(K,1),YC(K,1)	00006040
873	FORMAT(2X,F7.3,' ',F7.3,10X,F7.3,' ',F7.3,/))	00006050
874	FORMAT(2X,F7.3,' ',F7.3,10X,F7.3,' ',F7.3,/))	00006060
	AXD=AXX2.	00006070
	WRITE(6,875) AXD	00006080
875	FORMAT(' FASTENER HOLE DIAMETER = ',D9.3,' INCHES',/)	00006090
	ED=DABS(XC(K,3)-XC(K,1))/AXD	00006100
	WD=DABS(YC(K,3)-YC(K,1))/AXD	00006110
	WRITE(6,755) ED	00006120
755	FORMAT(' E/D RATIO = ',D9.3,/))	00006130
	WRITE(6,879) WD	00006140
879	FORMAT(' W/D RATIO = ',D9.3,/))	00006150
23	CONTINUE	00006160
C		00006170
C	PROCESS INPUT DATA ON PLATE GEOMETRIES	00006180
C		00006190
	WTH=YC(K,2)-YC(K,1)	00006200
	LM1=LOM	00006210
	CALL POLY(JK,K,XC,YC,W,AST,NCOL,LTNCH,BPR,IL)	00006220
	CALL CIRC(W,AST,JK,K,LTNCH,BPR,IL)	00006230
	IF(NOPT4.EQ.1.OR.NOPT4.EQ.3) CALL RCOUT(K)	00006240
	IF(NOPT4.EQ.2) CALL PSTRSS(K,LTNCH,BPR,IL)	00006250
	IF(NOPT4.EQ.4) CALL AVSTRS(K,LTNCH,NAVD,BPR,IL)	00006260
C		00006270
C	PERFORM FINITE GEOMETRY ANALYSIS FOR STRESS/DISPLACEMENT	00006280
C	STATE, COMPUTE FOUNDATION MODULI AND FAILURE VALUES	00006290
		00006300
	CALL FIOEOM(W,K,NOPT4,ITT)	00006310
	IF(BPR.NE.0.0.AND.BPR.NE.1.0.AND.IL.EQ.1) GO TO 21	00006320
	IF(BPR.NE.1.0.AND.LOM.E.1) CALL FBOLT(ANGK,H,K,NOPT1,LM1)	00006330
21	CALL FCRT(SALOM,H,WTH,AST,K,NOPT1,NOPT4,BPR,NAVD,IL)	00006340
20	CONTINUE	00006350
22	CONTINUE	00006360
	IF(BPR.EQ.1.0) GO TO 410	00006370
C		00006380
C	PREPARE INPUT FOR SEQUENTIAL PLY FAILURE	00006390
C	PREDICTION	00006400
		00006410
	IF(LOM.GT.1) GO TO 61	00006420
	N=NPLY(1)	00006430
	DO 30 I=1,N	00006440
	M=IPLY(I,1)	00006450
30	PLYK(I)=ANGX(M,1)	00006460
	N=NPLY(2)	00006470
	DO 60 I=1,N	00006480
	N1=I+NPLY(1)	00006490
	N2=IPLY(I,2)	00006500
60	PLYK(N1)=ANGK(N2,2)	00006510
61	CONTINUE	00006520
C		00006530
C		00006540
C	CALCULATION OF FASTENER STIFFNESSES.	00006550

C	FASQ=FASE/(2.*X(1.+FASV))	00006560
	FASLAM=5.*X(1.0+FASV)/(7.+5.*FASV)	00006570
	FASR=FASQ/2.	00006580
	FASA=ACOS(-1.)*FASR**2	00006590
	FASI=ACOS(-1.)*FASR**4/4.	00006600
	FASSS=FASLAM*FASQ*FASA	00006610
	FASBS=FASE*FASI	00006620
		00006630
CCC	INITIALIZATION	00006640
	IF(LDM.GT.1) GO TO 72	00006650
	ITT=0	00006660
	NTFL=0	00006670
	JNT=1	00006680
	P=0.	00006690
	DELP=1000.	00006700
	DO 5012 I=1,100	00006710
	NPNM(I,1)=I	00006720
	NPNM(I,2)=I+NPLY(1)	00006730
	UN(I)=0.	00006740
	QAMN(I)=0.	00006750
	MDAMP(I)=0.	00006760
	MDAMI(I)=0.	00006770
	PN(I)=0.	00006780
	BARK(I)=0.	00006790
5012	BARU(I)=0.	00006800
72	CONTINUE	00006810
		00006820
		00006830
	INCREMENTAL LOADS TO PLY FAILURE, PLY FAILURE	00006840
	MODES, AND FRACTIONAL STIFFNESS LOSSES ARE	00006850
	CALCULATED FOR EACH PLY FROM TOP TO BOTTOM	00006860
	UNTIL FINAL JOINT FAILURE	00006870
CCC		00006880
	90 CONTINUE	00006890
	ITT=ITT+1	00006900
	CALL CENTD(R,H,FASSS,FASBS,P,DELP,ITT)	00006910
	CALL SOLVE(U,H,P,DELP,NSDLS,ITT)	00006920
	CALL FAIL(QAMDLS,H,P,DELP,BPR,AST,WTH,PFAIL,ANGLE,NODE,	00006930
	*IROUT,NOPT4,NULTF,JNT,ITT,NTFL)	00006940
	CALL PRINT(U,P,DELP,PFAIL,ANGLE,BPR,NODE,IROUT,JNT,	00006950
	*NP,NSDLS,ITT)	00006960
	IF(JNT.EQ.0) GO TO 410	00006970
	IF(NLIM2.EQ.1) GO TO 90	00006980
	IF(NTFL.EQ.0.AND.NLIM2.GT.1) GO TO 90	00006990
71	CONTINUE	00007000
410	STOP	00007010
	END	00007020
		00007030
CCC		00007040
		00007050
		00007060
	SUBROUTINE STRTH(H,ES1,ES2,ESS,AF1,AF2,AF4,K)	00007070
	IMPLICIT REAL*8(A-H,O-Z)	00007080
	DIMENSION AINV(3,3),AVN(3),M(2),NV(3)	00007090
	DIMENSION NPLY(2),NUMPLY(2),ANG(3,2),IPLY(100,2)	00007100
	DIMENSION WK(25),PSMX(3),ES1(2),ES2(2),ESS(2)	00007110
	DIMENSION E1(2),E2(2),G12(2),V12(2),V21(2)	00007120
	COMMON/LYP/NPLY,NUMPLY,ANG,IPLY	00007130
	COMMON/MCD/E1,E2,G12,V12,V21	00007140
	COMMON/AM1/A	00007150

C		00007160
C	COMPUTE LAMINATE FAILURE LOADS BASED ON MAXIMUM	00007170
C	FIBER STRAINS FOR EACH FAILURE MODE	00007180
	CALL AMATRX(H,K)	00007190
	N=3	00007200
	IDOT=4	00007210
	IA=3	00007220
	CALL LINVZF(A,N,IA,AINV,IDOT,WK,IER)	00007230
	DO 100 KK=1,3	00007240
	DO 10 II=1,3	00007250
	NV(II)=0	00007260
10	AVN(II)=0.000	00007270
	IF(KK.EQ.1) NV(1)=1	00007280
	IF(KK.EQ.2) NV(1)=-1	00007290
	IF(KK.EQ.3) NV(3)=1	00007300
	DO 13 II=1,3	00007310
	DO 15 JJ=1,3	00007320
	AVN(IJ)=AVN(II)+AINV(IJ)*NV(JJ)	00007330
15	CONTINUE	00007340
	NP=NUMPLY(K)	00007350
	SMX=0.000	00007360
	RAD=DARCOS(-1.000)/180.000	00007370
	DO 25 II=1,NP	00007380
	TH=ANG(II,K)*RAD	00007390
	E11=DCOS(TH)*MM2+AVN(1)+AVN(2)*DSIN(TH)*MM2+	00007400
	*DCOS(TH)*DSIN(TH)*AVN(3)	00007410
	IF(KK.NE.1) GO TO 65	00007420
	EPRT=E11/ES2(K)	00007430
	GO TO 50	00007440
65	IF(KK.NE.2) GO TO 75	00007450
	EPRT=E11/ES1(K)	00007460
	GO TO 50	00007470
75	EPRT=E11/ES2(K)	00007480
50	CONTINUE	00007490
	IF(DABS(SMX).LT.DABS(EPRT)) SMX=EPRT	00007500
25	CONTINUE	00007510
	IF(DABS(SMX).GT.1.00-10) GO TO 555	00007520
	PSMX(KK)=ESS(K)*M012(K)	00007530
	GO TO 100	00007540
555	CONTINUE	00007550
	PSMX(KK)=DABS(1.000/SMX)	00007560
100	CONTINUE	00007570
	AF1=PSMX(1)	00007580
	AF2=PSMX(2)	00007590
	AF4=PSMX(3)	00007600
	RETURN	00007610
	END	00007620
		00007630
		00007640
		00007650
		00007660
	SUBROUTINE POLY(J,K,XC,YC,W,AST,NCOL,LTHCM,BPR,IL)	00007670
	IMPLICIT REAL*8(A-H,O-Z)	00007680
	DIMENSION XC(2,5),YC(2,5),A1(400),A2(400),XB(400)	00007690
	DIMENSION YB(400),T(400),A1A(4),A2A(4)	00007700
	COMMON/CMT1/XB,YB,A1,A2,T	00007710
		00007720
	ARRAY COLLOCATION POINTS AROUND EXTERIOR BOUNDARY	00007730
	AND APPLY STRESS BOUNDARY CONDITIONS	00007740
		00007750

	DO 120 I=1,4	00007760
	A1A(I)=0.	00007770
	A2A(I)=0.	00007780
120	CONTINUE	00007790
	W=DABS(YC(K,2)-YC(K,1))	00007800
	IF(LTNCM.EQ.1) A1A(1)=1000.0	00007810
	IF(LTNCM.EQ.2) A1A(1)=-1000.0	00007820
	IF(BPR.NE.0.0) A1A(3)=A1A(1)	00007830
	IF(IL.EQ.2) A1A(3)=0.0	00007840
	AST=DABS(A1A(1))	00007850
	J=0	00007860
	XC(K,5)=XC(K,1)	00007870
	YC(K,5)=YC(K,1)	00007880
	PI=DARCOS(-1.000)	00007890
	DAT=PI/NCOL	00007900
	DO 10 I=1,4	00007910
	X=XC(K,I)-XC(K,I+1)	00007920
	Y=YC(K,I+1)-YC(K,I)	00007930
	IF(X.EQ.0.) X=1.D-6	00007940
	IF(Y.EQ.0.) Y=1.D-6	00007950
	TH=DATAH2(X,Y)	00007960
	TH=TH*180./DARCOS(-0.101)	00007970
	DX=(XC(K,I+1)-XC(K,I))/(NCOL+1)	00007980
	DY=(YC(K,I+1)-YC(K,I))/(NCOL+1)	00007990
	DO 20 II=1,NCOL	00008000
	J=J+1	00008010
	IF(I.EQ.1.OR.I.EQ.4) GO TO 23	00008020
	YB(J)=YC(K,I)	00008030
	XB(J)=XC(K,I)+DX*(II-1/2)	00008040
	IF(II.EQ.1) XB(J)=XC(K,I)+(DX/2.)	00008050
	GO TO 24	00008060
23	CONTINUE	00008070
	IF(XC(K,5).NE.0.0) GO TO 26	00008080
	IF(I.NE.3) GO TO 24	00008090
	ADT=DAT*II	00008100
	XB(J)=YC(K,3)*DCOS((PI/2)+ADT)	00008110
	YB(J)=YC(K,3)*DSIN((PI/2)+ADT)	00008120
	TH=((PI/2)+ADT)*180./PI	00008130
	GO TO 24	00008140
26	CONTINUE	00008150
	YB(J)=YC(K,1)+DY*(II-1/2)	00008160
	IF(II.EQ.1) YB(J)=YC(K,1)+(DY/2.)	00008170
	XB(J)=XC(K,1)	00008180
24	T(J)=TH	00008190
	A1(J)=A1A(I)	00008200
	A2(J)=A2A(I)	00008210
20	CONTINUE	00008220
10	CONTINUE	00008230
	RETURN	00008240
	END	00008250
C		00008260
C		00008270
C		00008280
	SUBROUTINE CIRC(W,AST,JK,K,LTNCM,BPR,IL)	00008290
	IMPLICIT REAL*8(A-H,O-Z)	00008300
	DIMENSION X(400),Y(400),THTA(400),A1(400),A2(400)	00008310
	DIMENSION XB(400),YB(400)	00008320
	COMMON/FB1/BSTR,XSTR	00008330
	COMMON/CMT1/XB,YB,A1,A2,THTA	00008340
	COMMON/CMF2/X,Y,NPST,HAFT	00008350

C	COMMON/EL/A,B,N	00008360
C	ARRAY COLLOCATION POINTS AROUND INNER BOUNDARY	00008370
C	AND APPLY BEARING STRESS IN A COSINUSOIDAL	00008380
C	DISTRIBUTION	00008390
	CON=-1.0	00008400
	XSTR=AST	00008410
	BSTR=(2.*XMXSTR)/(DARCOS(CON)*B)	00008420
	IF(BPR.NE.0.0.AND.BPR.NE.1.0.AND.IE.EQ.1) BSTR=0.0	00008430
	IF(BPR.EQ.1.0) BSTR=0.	00008440
	NM4=N-4	00008450
	NQ=N*4/4	00008460
	DO 20 I=1,N	00008470
	JK=JK+1	00008480
	TH=((I-1)*2+1)*DARCOS(CON)/N	00008490
	X(I)=ANDCOS(TH)	00008500
	Y(I)=ANDSIN(TH)	00008510
	CS=-X(I)*BMB/(Y(I)*MAA)	00008520
	IF(Y(I).GT.0)THTA(JK)=DATAN(CS)-DARCOS(CON)/2.	00008530
	IF(Y(I).LT.0)THTA(JK)=DATAN(CS)+DARCOS(CON)/2.	00008540
	THTA(JK)=THTA(JK)*180./DARCOS(CON)	00008550
	IF(LTNCM.EQ.2) GO TO 25	00008560
	IF(I.GT.(NQ+1).AND.I.LT.(N-NQ)) GO TO 204	00008570
	GO TO 30	00008580
25	IF(I.LE.(NQ+2).OR.I.GE.(N-NQ-1)) GO TO 204	00008590
30	CONTINUE	00008600
	A1(JK)=0.	00008610
	A2(JK)=0.	00008620
	XB(JK)=X(I)	00008630
	YB(JK)=Y(I)	00008640
	GO TO 20	00008650
204	IF(Y(I).GT.0.) TETA=ARCCOS(-1.)-DATAN(Y(I)/X(I))	00008660
	IF(Y(I).LT.0.) TETA=ARCCOS(-1.)+DATAN(Y(I)/X(I))	00008670
	A1(JK)=-1.0*BSTR*NDABS(DCOS(TETA))	00008680
	A2(JK)=0.	00008690
	XB(JK)=X(I)	00008700
	YB(JK)=Y(I)	00008710
20	CONTINUE	00008720
	RETURN	00008730
	END	00008740
C	SUBROUTINE RCDUT(K)	00008750
C	SPECIFY COORDINATES AROUND CHARACTERISTIC CIRCLE	00008760
C	AN WHICH STRESSES ARE NEEDED FOR THE HOFFMAN/	00008770
C	TSAT-HILL FAILURE CRITERIA	00008780
C	IMPLICIT REAL*8(A-H,O-Z)	00008790
	DIMENSION X(400),Y(400),RCA(2),RCB(2),NRC(2)	00008800
	COMMON/CMT2/X,Y,NPST,NAST	00008810
	COMMON/RC/RCA,RCB,NRC	00008820
	COMMON/ELP/AX,BX,NDUT	00008830
	RAD=DARCOS(-0.1D1)/180.	00008840
	N1=NRC(K)	00008850
	DO 40 I=1,N1	00008860
	TINCR=360./NRC(K)	00008870
	THETA=(I-1)*TINCR*RAD	00008880
	C=DCOS(THETA)	00008890
	S=DSIN(THETA)	00008900
		00008910
		00008920
		00008930
		00008940
		00008950

```

K=DSQRT(1./((CM**2/RCA(K)**2)+(SM**2/RCB(K)**2))
X(I+NGOUT)=R*DCOS(THETA)
Y(I+NGOUT)=R*DSIN(THETA)
40 CONTINUE
RETURN
END

```

```

SUBROUTINE PSTRSS(K,NCS,BPR,IL)

```

```

SPECIFY DISCRETE COORDINATES OF POINTS AT WHICH
STRESSES ARE REQUIRED FOR THE POINT STRESS
CRITERION

```

```

IMPLICIT REAL*8(A-H,O-Z)
DIMENSION X(400),Y(400),DONT(2),DOBR(2)
DIMENSION DOSO(2),NPLY(2),NUMPLY(2),ANG(5,2)
DIMENSION IPLY(100,2),AONT(2,2),AOBR(2,2),AOSO(2,2)
COMMON/ELP/AX,BX,NGOUT
COMMON/CMT2/X,Y,NPST,NAST
COMMON/PSC1/DONT,DOBR,DOSO
COMMON/LYP/NPLY,NUMPLY,ANG,IPLY
COMMON/AOV/AONT,AOBR,AOSO
ANT=DONT(K)
ABR=DOBR(K)
ASO=DOSO(K)
IF(BPR.EQ.0.0.OR.BPR.L2.1.0) GO TO 25
ANT=AONT(IL,K)
ABR=AOBR(IL,K)
ASO=AOSO(IL,K)
5 CONTINUE
L=NGOUT+1
SQ=1.0
IF(NCS.EQ.1) SQ=-1.0
X(L)=0.
Y(L)=ANT+BX
X(L+1)=SQ*(AX+ABR)
Y(L+1)=0.
X(L+2)=SQ*(AX+ASO)
Y(L+2)=BX
NPST=3
DO 555 I=1,3
L=NGOUT+1+(I-1)
RETURN
END

```

```

SUBROUTINE AVSTRS(K,NCS,NAVD,BPR,IL)

```

```

SPECIFY COORDINATES OF POINTS ALONG WHICH
STRESSES WILL BE AVERAGED FOR THE AVERAGE
STRESS CRITERION

```

```

IMPLICIT REAL*8(A-H,O-Z)
DIMENSION X(400),Y(400),DONT(2),DOBR(2)
DIMENSION DOSO(2),NPLY(2),NUMPLY(2),ANG(5,2)
DIMENSION IPLY(100,2),AONT(2,2),AOBR(2,2),AOSO(2,2)
COMMON/AOV/AONT,AOBR,AOSO

```

```

00008960
00008970
00008980
00008990
00009000
00009010
00009020
00009030
00009040
00009050
00009060
00009070
00009080
00009090
00009100
00009110
00009120
00009130
00009140
00009150
00009160
00009170
00009180
00009190
00009200
00009210
00009220
00009230
00009240
00009250
00009260
00009270
00009280
00009290
00009300
00009310
00009320
00009330
00009340
00009350
00009360
00009370
00009380
00009390
00009400
00009410
00009420
00009430
00009440
00009450
00009460
00009470
00009480
00009490
00009500
00009510
00009520
00009530
00009540
00009550

```

COMMON/EL./AX,BX,NOUT	00009560
COMMON/CMT2/X,Y,NPST,NAST	00009570
COMMON/PSC1/DONT,DDBR,DOSO	00009580
COMMON/LYP/NPLY,NUMPLY,ANG,IPLY	00009590
ANT=DONT(K)	00009600
ABR=DDBR(K)	00009610
ASO=DOSO(K)	00009620
IF(BPR.EQ.0.0.OR.BPR.EQ.1.0) GO TO 23	00009630
ANT=ANT(IL,K)	00009640
ABR=ABR(IL,K)	00009650
ASO=ASO(IL,K)	00009660
25 CONTINUE	00009670
L=NOUT	00009680
SG=1.0	00009690
IF(NCS.EQ.1) SG=-1.0	00009700
ANDO=ANT/FLOAT(NAVD)	00009710
DO 20 I=1,NAVD	00009720
L=L+1	00009730
X(L)=0.	00009740
20 Y(L)=BX+ANDO/2.+(I-1)*ANDO	00009750
ANSO=ASO/FLOAT(NAVD)	00009760
DO 30 I=1,NAVD	00009770
L=L+1	00009780
X(L)=SG*(BX+ANSO/2.+(I-1)*ANSO)	00009790
30 Y(L)=BX	00009800
ANBR=ABR/FLOAT(NAVD)	00009810
DO 40 I=1,NAVD	00009820
L=L+1	00009830
X(L)=SO*(AX+ANBR/2.+(I-1)*ANBR)	00009840
40 Y(L)=0.	00009850
NAST=3*NAVD	00009860
N1=NOUT+1	00009870
N2=N1+NAST	00009880
NN=NOUT+3*NAVD	00009890
RETURN	00009900
END	00009910
	00009920
	00009930
SUBROUTINE FIEOM(H,KJ,NOPT4,ITT)	00009940
	00009950
	00009960
	00009970
	00009980
FIEOM PERFORMS A FINITE GEOMETRY ANALYSIS	00009990
USING THE BOUNDARY COLLOCATION TECHNIQUE	00010000
	00010010
	00010020
	00010030
IMPLICIT REAL*8(A-H,O-Z)	00010040
DIMENSION A(3,3),WK(25),AI(3,3),AZ(3),WKK(121),BC(400)	00010050
DIMENSION CH(4),H(2)	00010060
COMPLEX*16 GRHS(122)	00010070
COMPLEX*16 CM(196,124),CMC(196,121),CMCTCM(121,121),RHS(121)	00010080
COMMON/ROOTS/R1,R2	00010090
COMMON/TERMS/P1,Q1,P2,Q2	00010100
COMMON/ELP/AX,BX,NOUT	00010110
COMMON/SER/NT,NB	00010120
COMMON/AMT/A	00010130
COMPLEX*16 Z(4),Z1,Z2,Q1,Q2,P1,P2,R1,R2,WA(1488)	00010140
	00010150

C	AMATRX CALCULATES THE LAMINATE 'A' MATRIX	00010160
C	CALL AMATRX(H,KJ)	00010170
	I=3	00010180
	LDGT=4	00010190
	LA=3	00010200
C	LINEZF INVERTS THE 'A' MATRIX	00010210
C	CALL LINEZF(A,N,IA,AI,LDGT,HK,IER)	00010220
	NDEG=4	00010230
	AZ(1)=AI(1,1)/AI(2,2)	00010240
	AZ(2)=-2.*AI(1,3)/AI(2,2)	00010250
	AZ(3)=(2.*AI(1,2)+AI(3,3))/AI(2,2)	00010260
	AZ(4)=-2.*AI(2,3)/AI(2,2)	00010270
	AZ(5)=1.000	00010280
C	ZRPOLY FINDS THE ROOTS OF THE CHARACTERISTIC EQUATION	00010290
C	CALL ZRPOLY(AZ,NDEG,Z,IER)	00010300
C	Z(2) AND Z(4) ARE THE COMPLEX CONJUGATES OF Z(1)	00010310
C	AND Z(3) RESPECTIVELY	00010320
C	R1=Z(1)	00010330
C	R2=Z(3)	00010340
C	THE TWO ROOTS MUST BE CHECKED FOR A UNITARY COMPONENT	00010350
C	IN EITHER THE REAL OR IMAGINARY PART; SUCH AN	00010360
C	OCCURRENCE SIGNIFIES A QUASI-ISOTROPIC LAYUP AND	00010370
C	THE VALUE MUST BE PERTURBED SLIGHTLY IN ORDER TO	00010380
C	AVOID A SINGULAR MATRIX	00010390
	CH(1)=R1	00010400
	CH(2)=(0.0,-1.0)*R1	00010410
	CH(3)=R2	00010420
	CH(4)=(0.0,-1.0)*R2	00010430
	DO 30 IJK=1,4	00010440
	IF(DABS(CH(IJK)).LT.1.0D-10) CH(IJK)=1.0D-10	00010450
	AR=DABS(CH(IJK))	00010460
	IF(AR.LE.1.0) GO TO 11	00010470
	GO TO 32	00010480
31	IF((1.0-AR).LT.0.02) CH(IJK)=0.98	00010490
	GO TO 30	00010500
32	IF((AR-1.0).LT.0.02) CH(IJK)=1.02	00010510
30	CONTINUE	00010520
	R1=DCMPLX(CH(1),CH(2))	00010530
	R2=DCMPLX(CH(3),CH(4))	00010540
C	CONSTANTS P1,P2,Q1,Q2 ARE NEEDED FOR STRESS CALCULATIONS	00010550
C	P1=AI(1,1)*R1**2+AI(1,2)-AI(1,3)*R1	00010560
	P2=AI(1,1)*R2**2+AI(1,2)-AI(1,3)*R2	00010570
	Q1=AI(2,2)/R1+AI(1,2)*R1-AI(2,3)	00010580
	Q2=AI(2,2)/R2+AI(1,2)*R2-AI(2,3)	00010590
C	INPUTS AIN1(I),AIN2(I) ETC. REFER TO BOUNDARY CONDITIONS	00010600
C	NT4=4*NT	00010610
	NT8=8*NT	00010620
		00010630
		00010640
		00010650
		00010660
		00010670
		00010680
		00010690
		00010700
		00010710
		00010720
		00010730
		00010740
		00010750

```

NT8P4=8*NT+4
NT8P2=8*NT+2
NT8P1=8*NT+1
NB2=2*NB
NWK=NT8P1*(NT8P1+2)
CALL CMAT(BC,CMCTCM,CMC,CM,RHS,GRHS,NT4,NT8,NT8P4,NT8P2,
1NT8P1,NB2,NWK,WA,HKK,AI,NOPT4,KJ,ITT)
RETURN
END
00010760
00010770
00010780
00010790
00010800
00010810
00010820
00010830
00010840
00010850
00010860
00010870
00010880
00010890
00010900
00010910
00010920
00010930
00010940
00010950
00010960
00010970
00010980
00010990
00011000
00011010
00011020
00011030
00011040
00011050
00011060
00011070
00011080
00011090
00011100
00011110
00011120
00011130
00011140
00011150
00011160
00011170
00011180
00011190
00011200
00011210
00011220
00011230
00011240
00011250
00011260
00011270
00011280
00011290
00011300
00011310
00011320
00011330
00011340
00011350

SUBROUTINE AMATRX(H,K)
ASSEMBLE THE A MATRIX
IMPLICIT REAL*8(A-H,O-Z)
DIMENSION A(3,3),ANG(5,2),H(2),NPLY(2),NUMPLY(2)
DIMENSION E1(2),E2(2),Q12(2),V12(2),V21(2)
DIMENSION IPLY(100,2)
COMMON/MOD/E1,E2,Q12,V12,V21
COMMON/LYP/NPLY,NUMPLY,ANG,IPLY
COMMON/AMT/A
THKNES=NPLY(K)*H(K)
DENO=1.-E2(K)*V12(K)**2/E1(K)
Q11=E1(K)/DENO
Q22=E2(K)/DENO
Q12=V12(K)*Q22
Q21=Q12
Q33=Q12(K)
DO 10 I=1,3
DO 10 J=1,3
10 A(I,J)=0.
NN=NPLY(K)
DO 20 I=1,NN
T=H(K)
LP=IPLY(I,K)
THTAI=ANG(LP,K)*DARCOS(-1.D0)/180.D0
C=DCOS(THTAI)
S=DSIN(THTAI)
A(1,1)=(Q11*CN**4+2.*(Q12+2.*Q33)*CN*CMS+S+Q22*SN**4)*NT+A(1,1)
A(2,2)=(Q11*SN**4+2.*(Q12+2.*Q33)*CN*CMS+S+Q22*CN**4)*NT+A(2,2)
A(1,2)=(Q11+Q22-4.*Q33)*CN*CMS+S+Q12*(CN**4+SN**4)*NT+A(1,2)
A(2,1)=A(1,2)
A(3,3)=(Q11+Q22-2.*Q12-2.*Q33)*CN*CMS+S+Q33*(CN**4+SN**4)*NT+A(3,3)
A(1,3)=(Q11-Q12-2.*Q33)*CN*CMS+(Q12-Q22+2.*Q33)*SN**3*MC)*NT+A(1,3)
A(2,3)=(Q11-Q12-2.*Q33)*SN**3*MC+(Q12-Q22+2.*Q33)*CN**3*MS)*NT+A(2,3)
A(3,2)=A(2,3)
A(3,1)=A(1,3)
20 CONTINUE
DO 53 I=1,3
DO 53 J=1,3
A(I,J)=A(I,J)/THKNES
53 CONTINUE
RETURN
END
SUBROUTINE CMAT(BC,CMCTCM,CMC,CM,RHS,GRHS,NT4,NT8,NT8P4,NT8P2,

```

CCCCC

1HT8P1,NB2,NWK,WA,WKK,AMAT,NOPT4,KJ,ITT)

CMAT OUTPUTS STRESSES, STRAINS, AND DISPLACEMENTS
AT SPECIFIED COORDINATES

```

IMPLICIT REAL*8(A-H,O-Z)
DIMENSION RCA(2),RCB(2),NRCOUT(2)
DIMENSION ASX(400),ASXY(400)
COMMON/XXY1/ASX,ASXY
COMMON/ROOTS/R1,R2
COMMON/TERMS/P1,Q1,P2,Q2
COMMON/CMT1/X,Y,AIN1,AIN2,THTA
COMMON/CMT2/XOUT,YOUT,NPST,NAST
COMMON/FB2/FUR,FTHT,FMR
COMMON/GMT/RTHT,REPX,REPY,REPLY
COMMON/RC/RCA,RCB,NRCOUT
COMMON/ELP/AX,BX,HOUT
COMMON/SER/NT,NB
DIMENSION THTA(400),X(400),Y(400),AMAT(3,3)
DIMENSION AIN1(400),AIN2(400),BC(NB2)
DIMENSION XOUT(400),YOUT(400),WKK(NT8P1)
DIMENSION FUR(400),FTHT(400),FMR(400)
DIMENSION RTHT(400),REPX(400),REPY(400),REPLY(400)
COMPLEX*16 CMCTCM(NT8P1,NT8P1),RHS(NT8P1),PH1D,PHI2D,XETA1,XETA2
COMPLEX*16 CM(NB2,NT8P4),CMC(NB2,NT8P1),Z1,Z2,Z11,Z22,R1,R2
COMPLEX*16 T11,T12,T21,T22,P11,P12,P21,P22
COMPLEX*16 P1,P2,Q1,Q2,DCMPLX,CO,CSUM,GRHS(NT8P2)
COMPLEX*16 PH1DP,PHI2DP,PHI1DN,PHI2DN
COMPLEX*16 PH1P,PHI2P,PHI1N,PHI2N,PHI1,PHI2
COMPLEX*16 SV11,SV12,SV21,SV22,RB11,RB21,RB1B,RB2B
COMPLEX*16 R1B,R2B,P1B,P2B,Q1B,Q2B,WA(NWK)
A=AX
B=BX
CO=(0.0,1.0)
RB11=(Q1-P1MR1)/(A*CMR1NB)
RB21=(Q2-P2MR2)/(A*CMR2NB)
REALR1=R1
REALR2=R2
REALP1=P1
REALP2=P2
REALQ1=Q1
REALQ2=Q2
RRB11=RB11
RRB21=RB21
AIMGR1=CMR1
AIMGR2=CMR2
AIMGP1=CMP1
AIMGP2=CMP2
AIMGQ1=CMQ1
AIMGQ2=CMQ2
ARB11=CO*RB11
ARB21=CO*RB21
R1B=DCMPLX(REALR1,AIMGR1)
R2B=DCMPLX(REALR2,AIMGR2)
P1B=DCMPLX(REALP1,AIMGP1)
P2B=DCMPLX(REALP2,AIMGP2)
Q1B=DCMPLX(REALQ1,AIMGQ1)
Q2B=DCMPLX(REALQ2,AIMGQ2)

```

00011360
00011370
00011380
00011390
00011400
00011410
00011420
00011430
00011440
00011450
00011460
00011470
00011480
00011490
00011500
00011510
00011520
00011530
00011540
00011550
00011560
00011570
00011580
00011590
00011600
00011610
00011620
00011630
00011640
00011650
00011660
00011670
00011680
00011690
00011700
00011710
00011720
00011730
00011740
00011750
00011760
00011770
00011780
00011790
00011800
00011810
00011820
00011830
00011840
00011850
00011860
00011870
00011880
00011890
00011900
00011910
00011920
00011930
00011940
00011950

```

RB11B=DCMPLX(RRB11,ARB11) 00011960
RB21B=DCMPLX(RRB21,ARB21) 00011970
JJJ=0 00011980
DO 1000 I=1,NB 00011990
J=IM2 00012000
THTAI=THTA(I)*DARCOS(-1.0D0)/180.D0 00012010
C=DCOS(THTAI) 00012020
S=DSIN(THTAI) 00012030
P11=CXP1+SNQ1 00012040
P12=CXP2+SNQ2 00012050
P21=-SNP1+CXQ1 00012060
P22=-SNP2+CXQ2 00012070
T11=(CXCXR1*NR1+SN*2.*CXSXR1) 00012080
T12=(CXCXR2*NR2+SN*2.*CXSXR2) 00012090
T21=(-CXSXR1*NR1+CXS-(CXC-SN*NR1) 00012100
T22=(-CXSXR2*NR2+CXS-(CXC-SN*NR2) 00012110
Z1=X(I)+R1*Y(I) 00012120
Z2=X(I)+R2*Y(I) 00012130
Z11=CDSQRT(Z1*Z1-AXA-R1*NR1*NB) 00012140
Z22=CDSQRT(Z2*Z2-AXA-R2*NR2*NB) 00012150
REAL1=Z11 00012160
AIMO1=-CONZ11 00012170
IF(DABS(REAL1).LE.1.D-16)REAL1=0.0D0 00012180
IF(DABS(AIMO1).LE.1.D-16)AIMO1=0.0D0 00012190
Z11=DCMPLX(REAL1,AIMO1) 00012200
REAL2=Z22 00012210
AIMO2=-CONZ22 00012220
IF(DABS(REAL2).LE.1.D-16)REAL2=0.0D0 00012230
IF(DABS(AIMO2).LE.1.D-16)AIMO2=0.0D0 00012240
Z22=DCMPLX(REAL2,AIMO2) 00012250
XETA1=(Z1+Z11)/(A-COMR1*NB) 00012260
IF(CDABS(XETA1).LT.0.999) GO TO 300 00012270
GO TO 310 00012280
300 Z11=-Z11 00012290
XETA1=(Z1+Z11)/(A-COMR1*NB) 00012300
310 XETA2=(Z2+Z22)/(A-COMR2*NB) 00012310
IF(CDABS(XETA2).LT.0.999) GO TO 320 00012320
GO TO 330 00012330
320 Z22=-Z22 00012340
XETA2=(Z2+Z22)/(A-COMR2*NB) 00012350
330 CONTINUE 00012360
JJJ=JJJ+1 00012370
C 00012380
C 00012390
C 00012400
NORMAL & TANGENTIAL STRESS BOUNDARY CONDITIONS ARE IMPOSED 00012410
DO 5 N=1,NT 00012420
NP=N 00012430
CM(J-1,N)=NP*XETA1*NNPNT11/Z11 00012440
CM(J-1,2*NT+N)=NP*XETA2*NNPNT12/Z22 00012450
CM(J,N)=NP*XETA1*NNPNT21/Z11 00012460
CM(J,2*NT+N)=NP*XETA2*NNPNT22/Z22 00012470
NN=-N 00012480
CM(J-1,NT+N)=NN*XETA1*NNNT11/Z11 00012490
CM(J-1,3*NT+N)=NN*XETA2*NNNT12/Z22 00012500
CM(J,NT+N)=NN*XETA1*NNNT21/Z11 00012510
CM(J,3*NT+N)=NN*XETA2*NNNT22/Z22 00012520
5 CONTINUE 00012530
CM(J-1,NT8+1)=T11/Z11 00012540
CM(J-1,NT8+2)=T12/Z22 00012550
CM(J,NT8+1)=T21/Z11

```

	CM(I,NT8+2)=T22/Z22	0001256J
1300	CONTINUE	00012570
	DO 195 I=1,NB2	00012580
	DO 196 J=1,NT4	00012590
	REAL1=CM(I,J)	00012600
	AIM01=-CONJG(CM(I,J))	00012610
	IF(DABS(REAL1).LE.1.D-16)REAL1=0.000	00012620
	IF(DABS(AIM01).LE.1.D-16)AIM01=0.000	00012630
	CM(I,J)=DCMPLX(REAL1,AIM01)	00012640
	AIM02=-AIM01	00012650
	CM(I,NT4+J)=DCMPLX(REAL1,AIM02)	00012660
196	CONTINUE	00012670
195	CONTINUE	00012680
	DO 295 I=1,NB2	00012690
	DO 296 J=1,2	00012700
	REAL1=CM(I,NT8+J)	00012710
	AIM01=-CONJG(CM(I,NT8+J))	00012720
	IF(DABS(REAL1).LE.1.D-16)REAL1=0.000	00012730
	IF(DABS(AIM01).LE.1.D-16)AIM01=0.000	00012740
	CM(I,NT8+J)=DCMPLX(REAL1,AIM01)	00012750
	AIM02=-AIM01	00012760
	CM(I,NT8+2+J)=DCMPLX(REAL1,AIM02)	00012770
296	CONTINUE	00012780
295	CONTINUE	00012790
	SV11=(P2*Q1B-Q2*P1B)/(Q1*P2-Q2*P1)	00012800
	SV12=(P2*Q2B-Q2*P2B)/(Q1*P2-Q2*P1)	00012810
	SV21=(Q1*P1B-Q1B*P1)/(Q1*P2-Q2*P1)	00012820
	SV22=(Q1*P2B-Q2B*P1)/(Q1*P2-Q2*P1)	00012830
	DO 139 I=1,NB2	00012840
C	IMPOSE RIGID BODY ROTATION CONDITION	00012850
C	CM(I,2*NT+1)=-CM(I,1)*RB21/RB11+CM(I,2*NT+1)	00012860
	CM(I,4*NT+1)=-CM(I,1)*RB11B/RB11+CM(I,4*NT+1)	00012870
	CM(I,6*NT+1)=-CM(I,1)*RB21B/RB11+CM(I,6*NT+1)	00012880
	CM(I,1)=(0.0,0.0)	00012890
C	IMPOSE SINGLE-VALUEDNESS CONDITION	00012900
C	CM(I,NT8+3)=CM(I,NT8+1)*SV11+CM(I,NT8+3)	00012910
	CM(I,NT8+4)=CM(I,NT8+1)*SV12+CM(I,NT8+4)	00012920
	CM(I,NT8+3)=CM(I,NT8+2)*SV21+CM(I,NT8+3)	00012930
	CM(I,NT8+4)=CM(I,NT8+2)*SV22+CM(I,NT8+4)	00012940
	CM(I,NT8+1)=(0.0,0.0)	00012950
	CM(I,NT8+2)=(0.0,0.0)	00012960
139	CONTINUE	00012970
	DO 141 I=1,NB2	00012980
	DO 142 J=2,NT8	00012990
142	CM(I,J-1)=CM(I,J)	00013000
	CM(I,NT8)=CM(I,NT8+3)	00013010
	CM(I,NT8+1)=CM(I,NT8+4)	00013020
141	CONTINUE	00013030
	DO 95 I=1,NB2	00013040
	DO 96 J=1,NT8P1	00013050
	REAL1=CM(I,J)	00013060
	AIM01=-CONJG(CM(I,J))	00013070
	IF(DABS(REAL1).LE.1.D-16)REAL1=0.000	00013080
	IF(DABS(AIM01).LE.1.D-16)AIM01=0.000	00013090
	CM(I,J)=DCMPLX(REAL1,AIM01)	00013100
	AIM02=-AIM01	00013110
		00013120
		00013130
		00013140
		00013150

	CMC(I,J)=DCMPLX(REAL1,AIMG2)	00013150
96	CONTINUE	00013170
95	CONTINUE	00013180
	DO 120 I=1,NB	00013190
	J=IM2	00013200
	BC(J-1)=AIN1(I)	00013210
120	BC(J)=AIN2(I)	00013220
	DO 100 I=1,NT8P1	00013230
	DO 100 J=1,NT8P1	00013240
	CSUM=(0.0,0.0)	00013250
	DO 110 K=1,NB2	00013260
110	CSUM=CMC(K,I)*CM(K,J)+CSUM	00013270
	CMCTCM(I,J)=CSUM	00013280
100	CONTINUE	00013290
	DO 130 I=1,NT8P1	00013300
	CSUM=(0.0,0.0)	00013310
	DO 140 K=1,NB2	00013320
140	CSUM=CMC(K,I)*BC(K)+CSUM	00013330
130	RHS(I)=CSUM	00013340
	IJOB=0	00013350
	M=1	00013360
	CALL LEQ2C(CMCTCM,NT8P1,NT8P1,RHS,M,NT8P1,IJOB,WA,WKK,IER)	00013370
	QRHS(1)=-(RHS(2*NT)+RB21+RHS(4*NT)*RB11B+RHS(6*NT)*RB21B)/RB11	00013380
	QRHS(8*NT+1)=RHS(8*NT)*SV11+RHS(8*NT+1)*MSV12	00013390
	QRHS(8*NT+2)=RHS(8*NT)*SV21+RHS(8*NT+1)*MSV22	00013400
	DO 151 I=2,NT8	00013410
151	QRHS(I)=RHS(I-1)	00013420
C		00013430
C	STRESS AND STRAIN CALCULATION	00013440
C		00013450
	NRC3=NOUT+1	00013460
	IF(NOPT4.EQ.1.OR.NOPT4.EQ.3) NRCF=NOUT+NRCOUT(KJ)	00013470
	IF(NOPT4.EQ.2) NRCF=NOUT+NPST	00013480
	IF(NOPT4.EQ.4) NRCF=NOUT+NAST	00013490
	DO 190 K=1,NRCF	00013500
	Z1=XOUT(K)+R1*YOUT(K)	00013510
	Z2=XOUT(K)+R2*YOUT(K)	00013520
	Z11=CDSQRT(Z1*Z1-A*A-R1*R1*B*B)	00013530
	Z22=CDSQRT(Z2*Z2-A*A-R2*R2*B*B)	00013540
	XETA1=(Z1+Z11)/(A-C0*R1*B)	00013550
	IF(CDABS(XETA1).LT.0.999) GO TO 400	00013560
	GO TO 410	00013570
400	Z11=-Z11	00013580
	XETA1=(Z1+Z11)/(A-C0*R1*B)	00013590
410	XETA2=(Z2+Z22)/(A-C0*R2*B)	00013600
	IF(CDABS(XETA2).LT.0.999) GO TO 420	00013610
	GO TO 430	00013620
420	Z22=-Z22	00013630
	XETA2=(Z2+Z22)/(A-C0*R2*B)	00013640
430	CONTINUE	00013650
	PH11DP=(0.0,0.0)	00013660
	PH12DP=(0.0,0.0)	00013670
	PH11DN=(0.0,0.0)	00013680
	PH12DN=(0.0,0.0)	00013690
	PH11P=(0.0,0.0)	00013700
	PH12P=(0.0,0.0)	00013710
	PH11N=(0.0,0.0)	00013720
	PH12N=(0.0,0.0)	00013730
	DO 170 N=1,NT	00013740
	NP=N	00013750

	NN=-N	00013760
	PHI1DP=NP*(XETA1+NP*GRHS(N))/Z11+PHI1DP	00013770
	PHI1DN=NN*(XETA1+NP*GRHS(NT+N))/Z11+PHI1DN	00013780
	PHI2DP=NP*(XETA2+NP*GRHS(2*NT+N))/Z22+PHI2DP	00013790
	PHI2DN=NN*(XETA2+NP*GRHS(3*NT+N))/Z22+PHI2DN	00013800
	PHI1P=XETA1+NP*GRHS(N)+PHI1P	00013810
	PHI1N=XETA1+NP*GRHS(NT+N)+PHI1N	00013820
	PHI2P=XETA2+NP*GRHS(2*NT+N)+PHI2P	00013830
	PHI2N=XETA2+NP*GRHS(3*NT+N)+PHI2N	00013840
170	CONTINUE	00013850
	PHI1D=PHI1DP+PHI1DN+GRHS(8*NT+1))/Z11	00013860
	PHI2D=PHI2DP+PHI2DN+GRHS(8*NT+2))/Z22	00013870
	PHI1=PHI1P+PHI1N+GRHS(8*NT+1))*CDLOG(XETA1)	00013880
	PHI2=PHI2P+PHI2N+GRHS(8*NT+2))*CDLOG(XETA2)	00013890
	SOMAX=2.*(R1+R1*PHI1D+R2+R2*PHI2D)	00013900
	SOMAY=2.*(PHI1D+PHI2D)	00013910
	SOMAXY=-2.*(R1*PHI1D+R2*PHI2D)	00013920
	EPSX=AMAT(1,1)*SOMAX+AMAT(1,2)*SOMAY+AMAT(1,3)*SOMAXY	00013930
	EPSY=AMAT(2,1)*SOMAX+AMAT(2,2)*SOMAY+AMAT(2,3)*SOMAXY	00013940
	EPSXY=AMAT(3,1)*SOMAX+AMAT(3,2)*SOMAY+AMAT(3,3)*SOMAXY	00013950
	U=2.*(P1*PHI1+P2*PHI2)	00013960
	V=2.*(Q1*PHI1+Q2*PHI2)	00013970
	PI=DARCO(-1,D0)	00013980
	IF(XOUT(K).GT.0..AND.YOUT(K).GT.0.)	00013990
	+TETAA=DATAN(YOUT(K)/XOUT(K))*180./PI	00014000
	IF(XOUT(K).LT.0..AND.YOUT(K).GT.0.)	00014010
	+TETAA=DATAN(YOUT(K)/XOUT(K))*180./PI+180.	00014020
	IF(XOUT(K).LT.0..AND.YOUT(K).LT.0.)	00014030
	+TETAA=DATAN(YOUT(K)/XOUT(K))*180./PI+180.	00014040
	IF(YOUT(K).LT.0..AND.XOUT(K).GT.0.)	00014050
	+TETAA=DATAN(YOUT(K)/XOUT(K))*180./PI+360.	00014060
	C=DCOS(TETAA*PI/180.)	00014070
	S=DSIN(TETAA*PI/180.)	00014080
	SOMAR=C**2*SOMAX+S**2*SOMAY+2.*C*S*SOMAXY	00014090
	SOMAT=S**2*SOMAX+C**2*SOMAY-2.*C*S*SOMAXY	00014100
	SOMART=-C*S*SOMAX+C*S*SOMAY+(C**2-S**2)*SOMAXY	00014110
	EPSR=C**2*EPSX+S**2*EPSY+C*S*EPSXY	00014120
	EPSY=S**2*EPSX+C**2*EPSY-C*S*EPSXY	00014130
	EPSRT=2.*(C*S*EPSX+C*S*EPSY+(C**2-S**2)*(EPSXY/2.))	00014140
	UR=U+C*V*S	00014150
	RTHT(K)=TETAA	00014160
	REPX(K)=EPSX	00014170
	REPY(K)=EPSY	00014180
	REPLY(K)=EPSXY	00014190
	ASX(K)=SOMAX	00014200
	ASXY(K)=SOMAXY	00014210
	FUR(K)=UR	00014220
	FTHT(K)=TETAA	00014230
	FMR(K)=SOMAR	00014240
190	CONTINUE	00014250
	RETURN	00014260
	END	00014270
C		00014280
C		00014290
C		00014300
C	SUBROUTINE FBOLT(ANOK,H,K,NOPT1,LM1)	00014310
C		00014320
C		00014330
C	FBOLT CALCULATES THE INDIVIDUAL PLY FOUNDATION	00014340
	MODULI AND THE INDIVIDUAL PLY LOADS	00014350

```

IMPLICIT REAL*8(A-H,O-Z)
DIMENSION ATETAA(400),ANG(5,2),ASIGR(400),ASIGRT(400),H(2)
DIMENSION ASIG1(400),ASIG2(400),ASIG6(400),UR(400),ANGK(5,2)
DIMENSION FSMR(400),PLXPT(100)
DIMENSION IPLY(100,2),NPLY(2),NUMPLY(2)
DIMENSION FK1(100),PLX(100)
DIMENSION E11(2),E22(2),ESS(2),PMU12(2),PMU21(2)
DIMENSION RCA(2),RCB(2),NRC(2)
COMMON/STRESS/ASIGR,ASIGRT,ASIG1,ASIG2,ASIG6
COMMON/ELP/AX,BX,NOUT
COMMON/FB1/BSTR,XSTR
COMMON/LYP/NPLY,NUMPLY,ANG,IPLY
COMMON/FB2/UR,ATETAA,FSMR
COMMON/MD/E11,E22,ESS,PMU12,PMU21
COMMON/RC/RCA,RCB,NRC
COMMON/FCT/PLXPT
RAD=DARCOS(-0.101)/130.
THKTOT=NPLY(K)*H(K)
NN=NUMPLY(K)

```

CALCULATE DELEFF

```

WORK=0.
PLOADX=0.
IF(K.EQ.1) PLD=0.
DO 210 KK=1,NOUT
TH1=ATETAA(KK+1)-ATETAA(KK)
TH2=(ATETAA(KK)+ATETAA(KK+1))/2.
THETA=TH2*RAD
C=DCOS(THETA)
S=DSIN(THETA)
R=DSQRT(1./C**2/AX**2+S**2/BX**2)
FORCE=((FSMR(KK)+FSMR(KK+1))/2.)*R*TH1*RAD*THKTOT
WORK=WORK+FORCE*.5*((UR(KK)+UR(KK+1))/2.)
PLOADX=PLOADX+FORCE*C
210 CONTINUE
PLD=PLD+PLOADX
DELEFF=WORK/PLOADX

```

COMPUTE PLY STRESSES FROM LAMINATE STRAINS

```
(SIGMA)K,U,RO = (Q)*(EPS)R,O,RO
```

```

NN=NPLY(K)
DO 100 J=1,NN
LP=IPLY(J,K)
THETA=ANG(LP,K)*RAD
LI1=1
LI2=NOUT
NCAS=1
CALL QMATX(K,LI1,LI2,NCAS,NOPT1,RAD,THETA)

```

INTEGRATE AROUND CIRCULAR BOUNDARY FOR

```

00014360
00014370
00014380
00014390
00014400
00014410
00014420
00014430
00014440
00014450
00014460
00014470
00014480
00014490
00014500
00014510
00014520
00014530
00014540
00014550
00014560
00014570
00014580
00014590
00014600
00014610
00014620
00014630
00014640
00014650
00014660
00014670
00014680
00014690
00014700
00014710
00014720
00014730
00014740
00014750
00014760
00014770
00014780
00014790
00014800
00014810
00014820
00014830
00014840
00014850
00014860
00014870
00014880
00014890
00014900
00014910
00014920
00014930
00014940
00014950

```

C	INDIVIDUAL PLY LOADS AND COMPUTE FOUNDATION	00014960
CCCC	MODULI	00014970
		00014980
		00014990
	NNN=LI2-1	00015000
	PLOADX=0.	00015010
	WK=0.	00015020
	DO 70 I=LI1,NNN	00015030
	TH1=ATETAA(I+1)-ATETAA(I)	00015040
	TH2=(ATETAA(I)+ATETAA(I+1))/2.	00015050
	THETA=TH2*HRAD	00015060
	C=DCOS(THETA)	00015070
	S=DSIN(THETA)	00015080
	R=DSQRT(1/(C**2/AX**2+S**2/BX**2))	00015090
	FORCR=((ASIGR(I)+ASIGR(I+1))/2.)*R*TH1*HRAD*H(K)	00015100
	FORCRT=((ASIGRT(I)+ASIGRT(I+1))/2.)*R*TH1*HRAD*H(K)	00015110
	PLOADX=PLOADX+FORCR*C-FORCRT*S	00015120
70	CONTINUE	00015130
	FKI(I)=DABS(PLOADX/H(K)*DELEFF)	00015140
	PLX(I)=(K-1)*NPLY(1)+PLOADX	00015150
100	CONTINUE	00015160
	NT=NUMPLY(K)	00015170
	NN=NPLY(K)	00015180
	DO 310 I=1,NT	00015190
	DO 310 II=1,NN	00015200
	IF(IPLY(II,K).EQ.I) ANGK(I,K)=FKI(II)	00015210
	IF(IPLY(II,K).EQ.I) PLXPT(II)=PLX(II+(K-1)*NPLY(1))	00015220
310	CONTINUE	00015230
	NP=NUMPLY(K)	00015240
		00015250
		00015260
	COMPUTE TOTAL BEARING LOAD	00015270
		00015280
		00015290
	IF(K.EQ.1) GO TO 611	00015300
	PLXTOT=0.	00015310
	TH=H(1)*NPLY(1)+H(2)*NPLY(2)	00015320
	BLOAD=(BSTR*DAF*COS(-1.0D0)*8X*TH)/2.	00015330
	NN=NPLY(1)+NPLY(2)	00015340
	DO 212 I=1,NN	00015350
	PLXTOT=PLXTOT+PLX(I)	00015360
212	CONTINUE	00015370
611	CONTINUE	00015380
	RETURN	00015390
	END	00015400
		00015410
		00015420
		00015430
	SUBROUTINE FCRT(SALOH,H,UTH,AST,K,NOPT1,NOPT4,BPR,NAVD,IL)	00015440
		00015450
		00015460
	FAILURE LOAD CALCULATION	00015470
		00015480
		00015490
		00015500
	IMPLICIT REAL*8(A-H,O-Z)	00015510
	DIMENSION ASIG1(400),ASIG2(400),ASIG6(400),ASIGR(400)	00015520
	DIMENSION UR(400),FSMR(400),ATETAA(400),ASIGRT(400),NUMPLY(2)	00015530
	DIMENSION ANG(5,2),IPLY(100,2),HFMC(5,2),PLXPT(100),NPLY(2)	00015540
	DIMENSION PMS(5,2),PBR(5,2),P50(5,2),PALT(3,2)	00015550

	DIMENSION H(2), SALOW(2), SX(400), SXY(400), RCA(2), RCB(2), NRC(2,	00015560
	DIMENSION AEP51(400), PFL(5,2), PSTC(5,5,2)	00015570
	DIMENSION BPSTS(2,10,2,3)	00015580
	COMMON/BP1, BPSTS	00015590
	COMMON/RC/RCA, RCB, NRC	00015600
	COMMON/FB1/BSTR, XSTR	00015610
	COMMON/STRSS2/AEP51	00015620
	COMMON/FB2/UR, ATETAA, FSMR	00015630
	COMMON/FCT/FLXPT	00015640
	COMMON/ELP/AX, BX, NOUT	00015650
	COMMON/HFF/HFMC	00015660
	COMMON/LYP/NPLY, NUMPLY, ANG, IPLY	00015670
	COMMON/FAL1/PHS, PBR, P30, PALT	00015680
	COMMON/FAL3/PFL	00015690
	COMMON/STRESS/ASIOR, ASIORT, ASIO1, ASIO2, ASIO6	00015700
	COMMON/PSC2/PSTC	00015710
	COMMON/PSC3/SX, SXY	00015720
	RAD=DARCO3(-0.1D1)/180.	00015730
	IF(11.EQ.1.AND.(BPR.EQ.0.0.OR.BPR.EQ.1.0)) WRITE(6,89) K	00015740
59	FORMAT(///, ' ANALYSIS OF PLATE NO', I3, ' ', //)	00015750
	IF(NOPT4.EQ.2) GO TO 20	00015760
	IF(NOPT4.EQ.3) GO TO 90	00015770
	IF(NOPT4.EQ.4) GO TO 40	00015780
		00015790
	HOFFMAN/TSAI-HILL CRITERIA	00015800
		00015810
		00015820
	WRITE(6,10)	00015830
10	FORMAT(///, ' HOFFMAN/TSAI-HILL CRITERION', //)	00015840
	LI1=NOUT+1	00015850
	LI2=LI1+NRC(K)	00015860
	NCAS=2	00015870
	NN=NUMPLY(K)	00015880
	DO 402 I=1, NN	00015890
	THETA=ANG(I, K)*RAD	00015900
	CALL QMATX(K, LI1, LI2, NCAS, NOPT1, RAD, THETA)	00015910
	PFAIL=1.0D10	00015920
	N1=NRC(K)	00015930
	IF(BPR.EQ.0.0) DSB=DABS(PLXPT(I))	00015940
	IF(BPR.EQ.1.0) DSB=DABS(XSTR)	00015950
	DO 404 J=1, N1	00015960
	S1=ASIO1(J)/DSB	00015970
	S2=ASIO2(J)/DSB	00015980
	S6=ASIO6(J)/DSB	00015990
	CALL HOFF(S1, S2, S6, A, B, K)	00016000
	NNN=NOUT+J	00016010
		00016020
		00016030
		00016040
	FOR EACH PLY TYPE FIND THE LOCATION AND MAGNITUDE	00016050
	OF THE HIGHEST HOFFMAN/TSAI-HILL FAILURE INDEX VALUE	00016060
		00016070
		00016080
	P1=(-B+DSQRT(B**2+4*A))/(2.*A)	00016090
	P2=(-B-DSQRT(B**2+4*A))/(2.*A)	00016100
	IF(P1.LT.0.D0) PF=P2	00016110
	IF(P2.LT.0.D0) PF=P1	00016120
	IF(P1.LT.P2.AND.P1.GT.0.D0) PF=P1	00016130
	IF(P2.LT.P1.AND.P2.GT.0.D0) PF=P2	00016140
	IF(DABS(PF).GT.PFAIL) GO TO 480	00016150

```

PFAIL=DAB*(PF)
AMAX=A
BMAX=B
LOC=J
480 CONTINUE
404 CONTINUE
A=AMAX
B=BMAX
THTA=ATETAA(NOUT+LOC)

THE CORRESPONDING FAILURE LOAD IS OBTAINED FROM THE INDEX VALUE

XULT=HFMC(1,K)
IF(ASIO1(LOC).LT.0.) XULT=-HFIC(2,K)
YULT=HFMC(3,K)
IF(ASIC2(LOC).LT.0.) YULT=-HFMC(4,K)
SULT=HFIC(5,K)
IF(ASIO6(LOC).LT.0.) SULT=-HFMC(5,K)
SR1=ASIO1(LOC)/XULT
SR2=ASIO2(LOC)/YULT
SR6=ASIO6(LOC)/SULT
IF(BPR.EQ.0.0) WRITE(6,405) I,ANO(I,K),THTA,PFAIL,SR1,SR2,SR6
405 FORMAT(//,' FOR PLY TYPE NO ',I5,' ( ',D9.3,' DEGREES ),//,
* THE HIGHEST FAILURE INDEX WAS FOUND AT ',D9.3,' DEGREES',//,
* THE CORRESPONDING FAILURE LOAD = ',D9.3,' LBS',//,
* THE STRESS RATIOS AT THIS LOCATION ARE',//,
* SIG1/XULT = ',D9.3,/,
* SIG2/YULT = ',D9.3,/,
* SIG6/SULT = ',D9.3,/,
PFL(I,K)=PFAIL
402 CONTINUE
IF(BPR.EQ.0.0) GO TO 80
SFAIL=1.0D10
DO 110 I=1,NN
IF(SFAIL.GT.PFL(I,K)) NPY=I
110 IF(SFAIL.GT.PFL(I,K)) SFAIL=PFL(I,K)
PLFL=SFAIL*WTHM(K)*NPLY(K)
WRITE(6,771) PLFL
771 FORMAT(//,' FOR THE OPEN HOLE LAMINATE, FAILURE',//,
* IS PREDICTED AT A JOINT LOAD OF ',D9.3,' LBS',//)
GO TO 80
20 CONTINUE

POINT STRESS CRITERION

IF(IL.EQ.1.AND.(BPR.EQ.0.0.OR.BPR.EQ.1.0)) WRITE(6,50)
50 FORMAT(//,' POINT STRESS CRITERION ',//)
NN=NUMPLY(K)
NCAS=2
LI1=NOUT+1
LI2=LI1+2
IF(BPR.EQ.1.0.AND.NOPT1.EQ.2) NN=I
DO 100 I=1,NN
THETA=ANO(I,K)*RAD
CALL QMATX(K,LI1,LI2,NCAS,NOPT1,RAD,THETA)
IF(BPR.EQ.0.0.OR.BPR.EQ.1.0) GO TO 705

```

```

00016160
00016170
00016180
00016190
00016200
00016210
00016220
00016230
00016240
00016250
00016260
00016270
00016280
00016290
00016300
00016310
00016320
00016330
00016340
00016350
00016360
00016370
00016380
00016390
00016400
00016410
00016420
00016430
00016440
00016450
00016460
00016470
00016480
00016490
00016500
00016510
00016520
00016530
00016540
00016550
00016560
00016570
00016580
00016590
00016600
00016610
00016620
00016630
00016640
00016650
00016660
00016670
00016680
00016690
00016700
00016710
00016720
00016730
00016740
00016750

```

```

IF(IL.EQ.1) FAC=BPR
IF(IL.EQ.2) FAC=(1.-BPR)/PLXPT(I)
BPSTS(K,I,IL,1)=SX(1)*FAC/(AST*WTH*H(K)*NPLY(K))
BPSTS(K,I,IL,2)=SX(2)*FAC/(AST*WTH*H(K)*NPLY(K))
BPSTS(K,I,IL,3)=SXY(3)*FAC/(AST*WTH*H(K)*NPLY(K))
GO TO 100
705 CONTINUE
IF(BPR.EQ.0.0) DSN=DABS(PLXPT(I))
IF(BPR.EQ.1.0) DSN=XSTR
PNT=DSNMPSTC(1,I,K)/DABS(SX(1))
IF(SX(1).LT.0.) PNT=DSNMPSTC(2,I,K)/DABS(SX(1))
PBN=DSNMPSTC(3,I,K)/DABS(SX(2))
PSH=DSNMPSTC(4,I,K)/DABS(SXY(3))
IF(BPR.EQ.0.0) WRITE(6,70) I,ANG(I,K),PNT,PBN,PSH
70 FORMAT(/,' FOR PLY TYPE NUMBER ',I5,' WITH ',
* ' A PLY ORIENTATION OF ',D9.3,' DEGREES ',
* ' NET SECTION FAILURE LOAD = ',D9.3,' LBS ',
* ' BEARING FAILURE LOAD = ',D9.3,' LBS ',
* ' SHEAROUT FAILURE LOAD = ',D9.3,' LBS ',)
PNS(I,K)=PNT
PBR(I,K)=PBN
PSO(I,K)=PSH
100 CONTINUE
IF(BPR.EQ.0.0) GO TO 80
N=NPLY(K)
IF(BPR.EQ.1.0.AND.NOPT1.EQ.2) N=1
PFAIL1=1.0D10
PFAIL2=1.0D10
PFAIL3=1.0D10
DO 781 I=1,N
IF(PFAIL1.GT.PNS(I,K)) NPY1=I
IF(PFAIL1.GT.PNS(I,K)) PFAIL1=PNS(I,K)
IF(PFAIL2.GT.PBR(I,K)) NPY2=I
IF(PFAIL2.GT.PBR(I,K)) PFAIL2=PBR(I,K)
IF(PFAIL3.GT.PSO(I,K)) NPY3=I
781 IF(PFAIL3.GT.PSO(I,K)) PFAIL3=PSO(I,K)
IF(PFAIL1.GE.PFAIL2.OR.PFAIL1.GE.PFAIL3) GO TO 813
PFAIL1=PFAIL1*WTH*H(K)*NPLY(K)
WRITE(6,982) PFAIL1
WRITE(6,814)
GO TO 811
813 IF(PFAIL2.GE.PFAIL1.OR.PFAIL2.GE.PFAIL3) GO TO 812
PFAIL2=PFAIL2*WTH*H(K)*NPLY(K)
WRITE(6,982) PFAIL2
WRITE(6,815)
GO TO 811
812 IF(PFAIL3.GE.PFAIL1.OR.PFAIL3.GE.PFAIL2) GO TO 811
PFAIL3=PFAIL3*WTH*H(K)*NPLY(K)
WRITE(6,982) PFAIL3
WRITE(6,816)
811 CONTINUE
982 FORMAT(/,' FOR THE LAMINATE WITH AN OPEN HOLE, FAILURE ',
* ' IS PREDICTED AT A JOINT LOAD OF ',D9.3,' LBS ',)
814 FORMAT(' PREDICTED FAILURE MODE IS NET SECTION',)
815 FORMAT(' PREDICTED FAILURE MODE IS BEARING FAILURE',)
816 FORMAT(' PREDICTED FAILURE MODE IS SHEAR-OUT FAILURE',)
GO TO 80
C
C
C
AVERAGE STRESS CRITERION

```

```

40 CONTINUE
C
IF(IL.EQ.1.AND.(BPR.EQ.0.0.OR.BPR.EQ.1.0)) WRITE(6,55)
55 FORMAT(/, ' AVERAGE STRESS CRITERION ',/)
LI1=NOUT+1
NN=NUMPLY(K)
NCAS=2
IF(BPR.EQ.1.0.AND.NOPT1.EQ.2) NN=1
DO 105 I=1,NN
LI2=NOUT+3*NAVD
THETA=ANG(I,K)*RAD
CALL QMATX(K,LI1,LI2,NCAS,NOPT1,RAD,THETA)
C
CALCULATE AVERAGE STRESS
SUM=0.
N1=1
N2=NAVD
DO 200 IJ=N1,N2
200 SUM=SUM+SX(IJ)
AS1=SUM/FLOAT(NAVD)
N1=NAVD+1
N2=2*NAVD
SUM=0.
DO 215 IJ=N1,N2
215 SUM=SUM+SY(IJ)
AS2=SUM/FLOAT(NAVD)
SUM=0.
N1=2*NAVD+1
N2=3*NAVD
DO 220 IJ=N1,N2
220 SUM=SUM+SX(IJ)
AS3=SUM/FLOAT(NAVD)
IF(BPR.EQ.0.0.OR.BPR.EQ.1.0) GO TO 720
IF(IL.EQ.1) FAC=BPR
IF(IL.EQ.2) FAC=(1.-BPR)/PLXPT(1)
BPSTS(K,I,IL,1)=AS1*FAC/(ASTMWTM(K)*NPLY(K))
BPSTS(K,I,IL,2)=AS3*FAC/(ASTMWTM(K)*NPLY(K))
BPSTS(K,I,IL,3)=AS2*FAC/(ASTMWTM(K)*NPLY(K))
GO TO 105
720 CONTINUE
IF(BPR.EQ.0.0) DSN=DABS(PLXPT(1))
IF(BPR.EQ.1.0) DSN=XSTR
PNT=DSN*PSTC(1,I,K)/DABS(AS1)
IF(AS1.LT.0.) PNT=DSN*PSTC(2,I,K)/DABS(AS1)
PBN=DSN*PSTC(3,I,K)/DABS(AS3)
PSH=DSN*PSTC(4,I,K)/DABS(AS2)
IF(BPR.EQ.0.0) WRITE(6,75) I,ANG(I,K),PNT,PBN,PSH
75 FORMAT(/, ' FOR PLY TYPE NUMBER ',I5,' WITH ',/,
' A PLY ORIENTATION OF ',D3,' DEGREES ',/,
' NET SECTION FAILURE LOAD = ',D9.3,' LBS ',/,
' BEARING FAILURE LOAD = ',D9.3,' LBS ',/,
' SHEAROUT FAILURE LOAD = ',D9.3,' LBS ',/)
PNS(I,K)=PNT
PBR(I,K)=PBN
PSO(I,K)=PSH
105 CONTINUE
IF(BPR.EQ.0.0) GO TO 80
N=NUMPLY(K)

```

```

00017360
00017370
00017380
00017390
00017400
00017410
00017420
00017430
00017440
00017450
00017460
00017470
00017480
00017490
00017500
00017510
00017520
00017530
00017540
00017550
00017560
00017570
00017580
00017590
00017600
00017610
00017620
00017630
00017640
00017650
00017660
00017670
00017680
00017690
00017700
00017710
00017720
00017730
00017740
00017750
00017760
00017770
00017780
00017790
00017800
00017810
00017820
00017830
00017840
00017850
00017860
00017870
00017880
00017890
00017900
00017910
00017920
00017930
00017940
00017950

```



```

IF(BPR.EQ.1.0.AND.NOPT1.EQ.2) N=1
PFAIL1=1.0D10
PFAIL2=1.0D10
PFAIL3=1.0D10
DO 718 I=1,N
IF(PFAIL1.GT.PNS(I,K)) NPY1=I
IF(PFAIL1.GT.PNS(I,K)) PFAIL1=PNS(I,K)
IF(PFAIL2.GT.PBR(I,K)) NPY2=I
IF(PFAIL2.GT.PBR(I,K)) PFAIL2=PBR(I,K)
IF(PFAIL3.GT.PSO(I,K)) NPY3=I
718 IF(PFAIL3.GT.PSO(I,K)) PFAIL3=PSO(I,K)
IF(PFAIL1.GE.PFAIL2.OR.PFAIL1.GE.PFAIL3) GO TO 883
PFAIL1=PFAIL1*WTH*H(K)*NPLY(K)
WRITE(6,478) PFAIL1
WRITE(6,834)
GO TO 881
883 IF(PFAIL2.GE.PFAIL1.OR.PFAIL2.GE.PFAIL3) GO TO 882
PFAIL2=PFAIL2*WTH*H(K)*NPLY(K)
WRITE(6,473) PFAIL2
WRITE(6,835)
GO TO 881
882 IF(PFAIL3.GE.PFAIL1.OR.PFAIL3.GE.PFAIL2) GO TO 881
PFAIL3=PFAIL3*WTH*H(K)*NPLY(K)
WRITE(6,476) PFAIL3
WRITE(6,886)
881 CONTINUE
478 FORMAT(//,' FOR THE LAMINATE WITH THE OPEN HOLE, FAILURE',//,
* ' IS PREDICTED AT A JOINT LOAD OF ',D9.3,' LBS',//)
884 FORMAT(' PREDICTED FAILURE MODE IS NET SECTION',//)
885 FORMAT(' PREDICTED FAILURE MODE IS BEARING FAILURE',//)
886 FORMAT(' PREDICTED FAILURE MODE IS SHEAR-OUT FAILURE',//)
GO TO 80
C
C
C
90 CONTINUE
PAPP=XSTR*H(K)*NPLY(K)*WTH
WRITE(6,772)
772 FORMAT(//,' MAXIMUM STRAIN CRITERION ',//)
L11=NOUT+1
L12=L11+NRC(K)
NCAS=2
NN=NUMPLY(K)
DO 210 I=1,NN
THETA=ANG(I,K)*RAD
CALL GMATX(K,L11,L12,NCAS,NOPT1,RAD,THETA)
STMAX=-1.0D10
N1=NRC(K)
DO 510 J=1,N1
IF(STMAX.LT.DABS(AEPS1(J))) LOC=J
510 IF(STMAX.LT.DABS(AEPS1(J))) STMAX=DABS(AEPS1(J))
THTA=ATETAA(NOUT+LOC)
IF(BPR.EQ.1.0) GO TO 511
PFL(I,K)=DABS(PLXPT(I)*SALOW(K)/STMAX)
WRITE(6,774) I,ANG(I,K),THTA,PFL(I,K)
774 FORMAT(' FOR PLY TYPE NUMBER ',I5,' WITH ',//,
* ' A PLY ORIENTATION OF ',D9.3,' DEGREES ',//,
* ' FAILURE IS PREDICTED AT ',D9.3,' DEGREES ',//,
* ' AT A PLY LOAD OF ',D9.3,' LBS',//)
GO TO 210

```

```

00017960
00017970
00017980
00017990
00018000
00018010
00018020
00018030
00018040
00018050
00018060
00018070
00018080
00018090
00018100
00018110
00018120
00018130
00018140
00018150
00018160
00018170
00018180
00018190
00018200
00018210
00018220
00018230
00018240
00018250
00018260
00018270
00018280
00018290
00018300
00018310
00018320
00018330
00018340
00018350
00018360
00018370
00018380
00018390
00018400
00018410
00018420
00018430
00018440
00018450
00018460
00018470
00018480
00018490
00018500
00018510
00018520
00018530
00018540
00018550

```

```

511 PFL(I,K)=ABS(PAPP*SALOW(K)/STMAX)
210 CONTINUE
    IF(BPR.EQ.0.0) GO TO 80
    A=1.0D10
    NN=NUMPLY(K)
    DO 514 I=1,NN
        IF(A.GT.PFL(I,K)) NPY=I
514 IF(A.GT.PFL(I,K)) A=PFL(I,K)
    WRITE(6,778) A
778 FORMAT(//,' FOR THE OPEN HOLE LAMINATE, FAILURE IS',/,
* ' PREDICTED AT A JOINT LOAD OF ',D9.3,' LBS',//)
80 RETURN
    END

SUBROUTINE QMATX(K,LI1,LI2,NCAS,NOPT1,RAD,THETA)

QMATX PERFORMS BASIC STRESS AND STRAIN
TRANSFORMATIONS

IMPLICIT REAL*8(A-H,O-Z)
DIMENSION ASIGR(400),ASIGRT(400),ASIG1(400),ASIG2(400),ASIG6(400)
DIMENSION ATETAA(400),AEPSX(400),AEPSY(400),AEPSXY(400)
DIMENSION E11(2),E22(2),ESS(2),PMU12(2),PMU21(2),SX(400),SXY(400)
DIMENSION AEPS1(400)
DIMENSION ASX(400),ASXY(400)
COMMON/XXY1/ASX,ASXY
COMMON/MOD/E11,E22,ESS,PMU12,PMU21
COMMON/STRSS2/AEPS1
COMMON/STRESS/ASIGR,ASIGRT,ASIG1,ASIG2,ASIG6
COMMON/QMT/ATETAA,AEPSX,AEPSY,AEPSXY
COMMON/PSC3/SX,SXY
J=0
Q11=E11(K)/(1.0-PMU12(K)*PMU21(K))
Q12=(PMU21(K)*E11(K))/(1.0-PMU12(K)*PMU21(K))
Q22=E22(K)/(1.0-PMU12(K)*PMU21(K))
Q66=ESS(K)
C=DCOS(THETA)
S=DSIN(THETA)
BQ11=(Q11*(C**4))+(2.*(Q12+(2.*Q66))*(C**2)*(S**2))+(Q22*(S**4))
BQ12=((Q11+Q22-(4.*Q66))*(S**2)*(C**2))+(Q12*(S**4+C**4))
BQ16=((Q11-Q12-(2.*Q66))*(S*(C**3)))+(Q12-Q22+(2.*Q66))*(S**3)*C
*)
BQ22=(Q11*(S**4))+(2.*(Q12+(2.*Q66))*(S**2)*(C**2))+(Q22*(C**4))
BQ26=((Q11-Q12-(2.*Q66))*C*(S**3))+(Q12-Q22+(2.*Q66))*S*(C**3)
BQ66=((Q11+Q22-(2.*(Q12+Q66)))*(S**2)*(C**2))+(Q66*(C**4)+(S**4))
*)
DO 40 I=LI1,LI2
    J=J+1
    IF(NCAS.EQ.1) THETA=ATETAA(I)*RAD
    C=DCOS(THETA)
    S=DSIN(THETA)
    SIGX=BQ11*AEPSX(I)+BQ12*AEPSY(I)+BQ16*AEPSXY(I)
    SIGY=BQ12*AEPSX(I)+BQ22*AEPSY(I)+BQ26*AEPSXY(I)
    SIGXY=BQ16*AEPSX(I)+BQ26*AEPSY(I)+BQ66*AEPSXY(I)
    SX(J)=SIGX
    SXY(J)=SIGXY
    IF(NOPT1.EQ.2) SX(J)=ASX(I)
    IF(NOPT1.EQ.2) SXY(J)=ASXY(I)

```

```

00018560
00018570
00018580
00018590
00018600
00018610
00018620
00018630
00018640
00018650
00018660
00018670
00018680
00018690
00018700
00018710
00018720
00018730
00018740
00018750
00018760
00018770
00018780
00018790
00018800
00018810
00018820
00018830
00018840
00018850
00018860
00018870
00018880
00018890
00018900
00018910
00018920
00018930
00018940
00018950
00018960
00018970
00018980
00018990
00019000
00019010
00019020
00019030
00019040
00019050
00019060
00019070
00019080
00019090
00019100
00019110
00019120
00019130
00019140
00019150

```

```

ASIGR(I)=SIGX*CM2+SIGY*SM2+2.*SIGXY*SMC      00019160
ASIGRT(I)=-SIGX*SMC+SIGY*CM2+SIGXY*(CM2-SM2)  00019170
MSIG1(J)=SIGX*CM2+SIGY*SM2+2.*SMC*SIGXY      00019180
ASIG2(J)=SIGX*SM2+SIGY*CM2-2.*SMC*SIGXY      00019190
ASIG6(J)=-CM2*SIGX+SIGY*SM2+(CM2-SM2)*SIGXY  00019200
AEPS1(J)=AEPSX(I)*CM2+AEPSY(I)*SM2+AEPSXY(I)*SMC 00019210
40 CONTINUE      00019220
RETURN          00019230
END            00019240
00019250
00019260
00019270
SUBROUTINE HOFF(S1,S2,S6,A,B,K)      00019280
IMPLICIT REAL*8(A-H,O-Z)          00019290
DIMENSION HPMC(5,2)                00019300
COMMON/HFF/HPMC                    00019310
00019320
00019330
00019340
COMPUTE THE HOFFMAN/TSAL-HILL FAILURE INDEX  00019350
00019360
00019370
A=0.0D0      00019380
B=0.0D0      00019390
XC=HPMC(1,K) 00019400
XT=HPMC(2,K) 00019410
YC=HPMC(3,K) 00019420
YT=HPMC(4,K) 00019430
STC=HPMC(5,K) 00019440
A=(S1*S2-S1*XT)/(XC*XT)+(S2*S2)/(YC*YT)+(S6/STC)*SM2 00019450
IF(XC.EQ.XT.AND.YC.EQ.YT) GO TO 10 00019460
B=((XC-XT)/(XC*XT))*S1+((YC-YT)/(YC*YT))*S2 00019470
GO TO 20      00019480
10 CONTINUE   00019490
B=0.0D0      00019500
20 CONTINUE   00019510
RETURN        00019520
END           00019530
00019540
00019550
00019560
00019570
SUBROUTINE CENTD(RF,H,FASSS,FASBS,P,DELP,ITT) 00019580
00019590
00019600
IMPLICIT REAL*8(A-H,O-Z)          00019610
DIMENSION PLYK(100),BARK(100),BARU(100),F(100) 00019620
DIMENSION H(2),RF(2)              00019630
DIMENSION AII(100,100),A(2),B(2)  00019640
DIMENSION NPLY(2)                  00019650
COMMON/PBB/PLYK,BARK,BARU          00019660
COMMON/AFM/AII,F                    00019670
COMMON/LYP/NPLY                     00019680
00019690
00019700
SET UP THE CENTRAL DIFFERENCE EQUATIONS  00019710
00019720
DO 3 I=1,100      00019730
DO 3 J=1,100      00019740
3 AII(I,J)=0.      00019750
00019760
NECESSARY CONSTANTS ARE FORMED

```

C	DO 7 I=1,2	00019750
	A(I)=H(I)*H2/FASSS	00019770
7	B(I)=H(I)*H4/FASBS	00019780
	H12=H(1)/H(2)	00019790
	A1=H(1)*H2/FASSS	00019800
	A2=H(2)*H2/FASSS	00019810
	NP=NPLY(1)+NPLY(2)	00019820
		00019830
C		00019840
C		00019850
C	SHEAR AT TOP OF JOINT EQUALS ZERO	00019860
	AII(1,1)=1.	00019870
	AII(1,2)=-2.+A1*PLYK(2)	00019880
	AII(1,4)=2.+A1*PLYK(2)	00019890
	AII(1,5)=-1.	00019900
	F(1)=0.0	00019910
C		00019920
C		00019930
C	MOMENT CONDITION AT TOP	00019940
	IF(RF(1).GE.1.D10) GO TO 50	00019950
	Z=1.	00019960
	R=RF(1)	00019970
	GO TO 60	00019980
50	Z=0.	00019990
	R=1.	00020000
60	AII(2,1)=R	00020010
	AII(2,2)=(Z*H2.*H(1)*FASSS)+R*(-2.-A1*PLYK(2)+(H(1)*H2	00020020
	*FASSS)/FASBS)	00020030
	AII(2,3)=-Z*(4.*H(1)*FASSS+(2*H(1)*H2*PLYK(1)*H(1)))	00020040
	AII(2,4)=Z*H2.*H(1)*FASSS+R*(2.+A1*PLYK(2)-(H(1)*H2	00020050
	*FASSS)/FASBS)	00020060
	AII(2,5)=-R	00020070
	F(2)=Z*H2.*H(1)*H3*BARK(1)*BARU(1)	00020080
C		00020090
C		00020100
C	GOVERNING EQUATIONS FOR THE TOP PLATE	00020110
	H2=NPLY(1)	00020120
	DO 55 J=1,N2	00020130
	I=J+2	00020140
	AII(I,J)=1.	00020150
	IF(J.EQ.1) GO TO 56	00020160
	AII(I,J+1)=-4.-A(1)*PLYK(J-1)	00020170
	GO TO 57	00020180
56	AII(I,J+1)=-4.-A(1)*PLYK(2)	00020190
57	AII(I,J+2)=6.+(2.*A(1)+B(1))*PLYK(J)	00020200
	IF(J.EQ.N2) GO TO 61	00020210
	AII(I,J+3)=-4.-A(1)*PLYK(J+1)	00020220
	GO TO 62	00020230
61	AII(I,J+3)=-4.-A(1)*PLYK(NPLY(1)-1)	00020240
62	AII(I,J+4)=1.	00020250
	IF(J.EQ.1) GO TO 58	00020260
	IF(J.EQ.N2) GO TO 63	00020270
	F(1)=A(1)*BARK(J-1)*BARU(J-1)	00020280
	M=-(2.*A(1)+B(1))*BARK(J)*BARU(J)	00020290
	M+A(1)*BARK(J+1)*BARU(J+1)	00020300
	GO TO 59	00020310
58	F(1)=2.*A(1)*BARK(2)*BARU(2)	00020320
	M=-(2.*A(1)+B(1))*BARK(1)*BARU(1)	00020330
	GO TO 59	00020340
		00020350

	63 F(I)=2.*H(1)*MBARK(NPLY(1)-1)*MBARU(NPLY(1)-1)	00020360
	X=(2.*A(1)+B(1))*BARK(J)*BARU(J)	00020370
	59 CONTINUE	00020380
	55 CONTINUE	00020390
C	INTERFACE SHEAR ON TOP PLATE = P+DELP	00020400
C	I=NPLY(1)+3	00020410
C	J=NPLY(1)	00020420
	AII(I,J)=1.	00020430
	AII(I,J+1)=-((2.+A1*PLYK(NPLY(1)-1))	00020440
	AII(I,J+3)=2.+A1*PLYK(NPLY(1)-1)	00020450
	AII(I,J+4)=-1.	00020460
	F(I)=(-2.*H(1)*X3*(P+DELP))/FASBS	00020470
C	SLOPE CONTINUITY	00020480
C	I=NPLY(1)+4	00020490
C	J=NPLY(1)	00020500
	AII(I,J)=1.	00020510
	AII(I,J+1)=-((2.+A1*PLYK(NPLY(1)-1)-H(1)*X2*FASBS)/FASBS)	00020520
	AII(I,J+3)=2.+A1*PLYK(NPLY(1)-1)-H(1)*X2*FASBS/FASBS	00020530
	AII(I,J+4)=-1.	00020540
	AII(I,J+5)=-H12*X3	00020550
	AII(I,J+6)=H12*X3*(2.+A2*PLYK(NPLY(1)+2)-H(2)*X2*FASBS/FASBS)	00020560
	AII(I,J+8)=-H12*X3*(2.+A2*PLYK(NPLY(1)+2)-H(2)*X2*FASBS/FASBS)	00020570
	AII(I,J+9)=H12*X3	00020580
	F(I)=0.	00020590
C	MOMENT CONTINUITY	00020600
C	I=NPLY(1)+5	00020610
C	J=NPLY(1)+1	00020620
	AII(I,J)=1.	00020630
	AII(I,J+1)=-((2.+A1*PLYK(NPLY(1)))	00020640
	AII(I,J+2)=1.	00020650
	AII(I,J+5)=-H12*X2	00020660
	AII(I,J+6)=H12*X2*(2.+A2*PLYK(NPLY(1)+1))	00020670
	AII(I,J+7)=-H12*X2	00020680
	F(I)=A1*(BARK(NPLY(1))*BARU(NPLY(1))-BARK(NPLY(1)+1)*	00020690
	MBARU(NPLY(1)+1))	00020700
C	INTERFACE SHEAR ON BOTTOM PLATE	00020710
C	I=NPLY(1)+6	00020720
C	J=NPLY(1)+5	00020730
	AII(I,J)=-1.	00020740
	AII(I,J+1)=(2.+A2*PLYK(NPLY(1)+2))	00020750
	AII(I,J+3)=-((2.+A2*PLYK(NPLY(1)+2))	00020760
	AII(I,J+4)=1.	00020770
	F(I)=2.*H(2)*X3*(P+DELP)/FASBS	00020780
C	GOVERNING EQUATIONS FOR THE BOTTOM PLATE	00020790
C	N1=NPLY(1)+7	00020800
	N2=NPLY(1)+NPLY(2)+6	00020810
	DO 70 I=N1,N2	00020820
	J=I-2	00020830
	AII(I,J)=1.	00020840
	IF(I.EQ.N1) GO TO 71	00020850
		00020860
		00020870
		00020880
		00020890
		00020900
		00020910
		00020920
		00020930
		00020940
		00020950

AII(I,J+1)=-4.-A(2)*PLYK(J-5)	00020960
GO TO 72	00020970
71 AII(I,J+1)=-4.-A(2)*PLYK(NPLY(1)+2)	00020980
72 AII(I,J+2)=6.+(2.*A(2)+B(2))*PLYK(J-4)	00020990
IF(I.EQ.N2) GO TO 75	00021000
AII(I,J+3)=-4.-A(2)*PLYK(J-3)	00021010
GO TO 76	00021020
75 AII(I,J+3)=-4.-A(2)*PLYK(J-5)	00021030
76 AII(I,J+4)=1.	00021040
IF(I.EQ.N1) GO TO 73	00021050
IF(I.EQ.N2) GO TO 77	00021060
F(I)=A(2)*BARK(J-5)*BARU(J-5)	00021070
M=(2.*A(2)+B(2))*BARK(J-4)*BARU(J-4)	00021080
M+A(2)*BARK(J-3)*BARU(J-3)	00021090
GO TO 74	00021100
73 F(I)=2.*A(2)*BARK(NPLY(1)+2)*BARU(NPLY(1)+2)	00021110
M=(2.*A(2)+B(2))*BARK(J-4)*BARU(J-4)	00021120
GO TO 74	00021130
77 F(I)=2.*A(2)*BARK(J-5)*BARU(J-5)	00021140
M=(2.*A(2)+B(2))*BARK(J-4)*BARU(J-4)	00021150
74 CONTINUE	00021160
70 CONTINUE	00021170
SHEAR ON BOTTOM PLATE EQUALS ZERO	
NP=NPLY(1)+NPLY(2)	00021180
I=NP+7	00021190
J=NP+4	00021200
AII(I,J)=-1.	00021210
AII(I,J+1)=(2.+A2*PLYK(NP-1))	00021220
AII(I,J+3)=-2.+A2*PLYK(NP-1)	00021230
AII(I,J+4)=1.	00021240
F(I)=0.	00021250
MOMENT BOUNDARY CONDITION ON BOTTOM PLATE	
I=NP+8	00021260
IF(RF(2).OE.1.D10) GO TO 85	00021270
Z=1.	00021280
R=RF(2)	00021290
GO TO 95	00021300
85 Z=0.	00021310
R=1.	00021320
95 AII(I,J)=-R	00021330
AII(I,J+1)=Z*(2.*H(2)*FASSS)+R*(2.+A2*PLYK(NP-1))	00021340
M=H(2)*H2*FASSS/FASBS	00021350
AII(I,J+2)=-Z*(4.*H(2)*FASSS+2.*H(2)*H3*PLYK(NP))	00021360
AII(I,J+3)=Z*2.*H(2)*FASSS+R*(-2.-A2*PLYK(NP-1))	00021370
M+H(2)*H2*FASSS/FASBS	00021380
AII(I,J+4)=R	00021390
F(I)=Z*(2.*H(2)*H3*BARK(NP)*BARU(NP))	00021400
RETURN	00021410
END	00021420
SUBROUTINE SOLVE(U,H,P,DELP,NSDLS,ITT)	
	00021430
	00021440
	00021450
	00021460
	00021470
	00021480
	00021490
	00021500
	00021510
	00021520
	00021530
	00021540
	00021550

	IMPLICIT REAL*8(A-H,O-Z)	00021560
	DIMENSION A(100,100),B(100),NPLY(2),U(100),F(100)	00021570
	DIMENSION SX(100),PLYK(100),H(2)	00021580
	DIMENSION BARK(100),BARU(100)	00021590
	COMMON/LYP/NPLY	00021600
	COMMON/AFM/A,F	00021610
	COMMON/PBB/PLYK,BARK,BARU	00021620
C		00021630
C	SOLUTION OF THE SYSTEM: 9AI(U)=(B)	00021640
	NP=NPLY(1)+NPLY(2)+8	00021650
	DO 444 I=1,NP	00021660
444	B(I)=F(I)	00021670
		00021680
		00021690
		00021700
C		00021710
C		00021720
C	APPLYING GUASSIAN ELIMINATION TO THE	00021730
C	MATRIX OF COEFFICIENTS	00021740
	DO 2001 I=1,NP	00021750
	IR=I	00021760
2042	IF(A(IR,I).NE.0.) GO TO 2041	00021770
	IR=IR+1	00021780
	IF(IR.GT.NP) GO TO 2001	00021790
	GO TO 2042	00021800
2041	NN=IR+1	00021810
	DO 2002 L=NN,NP	00021820
	IF(DABS(A(L,I)).GT.1.D-30) GO TO 2009	00021830
	A(L,I)=0.	00021840
	GO TO 2002	00021850
2009	CF=-A(IR,I)/A(L,I)	00021860
	DO 2003 J=I,NP	00021870
	A(L,J)=A(L,J)+CF+A(IR,J)	00021880
	IF(DABS(A(L,J)).LT.1.D-30) A(L,J)=0.	00021890
2003	CONTINUE	00021900
	B(L)=B(L)+CF+B(I)	00021910
2002	CONTINUE	00021920
2001	CONTINUE	00021930
C		00021940
C	BACK SUBSTITUTION	00021950
	DO 2011 I=1,NP	00021960
	L=NP+1-I	00021970
	SUM=0.	00021980
	IF(A(L,L).EQ.0.) GO TO 2112	00021990
	N=L+1	00022000
	IF(N.GT.NP) GO TO 2013	00022010
	DO 2013 J=N,NP	00022020
	SUM=SUM-A(L,J)*SX(J)	00022030
2013	CONTINUE	00022040
	SX(L)=(B(L)+SUM)/A(L,L)	00022050
	GO TO 2011	00022060
2112	CONTINUE	00022070
	SX(L)=0.	00022080
2011	CONTINUE	00022090
C		00022100
C	EQUILIBRIUM CHECK	00022110
		00022120
		00022130
		00022140
	NPTS=NPLY(1)+NPLY(2)+8	00022150

	PT=P+DELP	00022160
	N1=NPLY(1)+2	00022170
	N2=NPLY(1)+7	00022180
	NN=NPLY(1)+NPLY(2)+6	00022190
	SUM4=0.	00022200
	SUM5=0.	00022210
	DO 1444 I=3,N1	00022220
	J=I-2	00022230
	U(J)=SX(I)	00022240
	SUM4=SUM4+SX(I)*PLYK(J)*H(1)	00022250
1444	CONTINUE	00022260
	DO 1555 I=N2,NN	00022270
	J=I-6	00022280
	U(J)=SX(I)	00022290
	SUM5=SUM5+SX(I)*PLYK(J)*H(2)	00022300
1555	CONTINUE	00022310
	IF(NSDLS.EQ.1) GO TO 810	00022320
	PT=PT*2.	00022330
	SUM4=SUM4*2.	00022340
	SUM5=SUM5*2.	00022350
810	CONTINUE	00022360
	NP=NPLY(1)+NPLY(2)	00022370
	N=NPLY(1)+NPLY(2)	00022380
	II=1	00022390
	DO 311 I=1,N	00022400
	IF(I.GT.NPLY(1)) II=2	00022410
	PL=U(I)*PLYK(I)*H(II)	00022420
	IF(I.LE.NPLY(1)) GO TO 311	00022430
311	CONTINUE	00022440
	RETURN	00022450
	END	00022460
C		00022470
C		00022480
	SUBROUTINE FAIL(GAMD, U, H, P, DELP, BPR, AST, WTH, PFAIL, ANGLE, NODE,	00022490
	MIROUT, NOPT4, NULYF, JNT, ITT, NTFL)	00022500
C		00022510
C		00022520
	IMPLICIT REAL*8(A-H,O-Z)	00022530
	DIMENSION NPLY(2), MDAMP(100), H(2), PLYK(100), U(100)	00022540
	DIMENSION BARK(100), BARU(100)	00022550
	DIMENSION PN(100), MDAMI(100), GAMD(2), GAMN(100)	00022560
	DIMENSION DELNS(5,2), DELBR(5,2), DELSO(5,2)	00022570
	DIMENSION UN(100), PFL(5,2), PSTC(5,5,2)	00022580
	DIMENSION IPLY(100,2), ANG(5,2), NUMPLY(2)	00022590
	DIMENSION PNS(5,2), PBR(5,2), PSO(5,2), PALT(3,2)	00022600
	DIMENSION BPSTS(2,10,2,3)	00022610
	DIMENSION NPNM(100,2)	00022620
	COMMON/COUNT/NPNM	00022630
	COMMON/BP1/BPSTS	00022640
	COMMON/PSC2/PSTC	00022650
	COMMON/FAL1/PNS, PBR, PSO, PALT	00022660
	COMMON/FAL3/DELNS, DELBR, DELSO	00022670
	COMMON/FAL4/UN, GAMN, MDAMP, MDAMI, PN	00022680
	COMMON/FAL5/PFL	00022690
	COMMON/PBB/PLYK, BARK, BARU	00022700
	COMMON/PRT/NDAM, INPLY, ITYP	00022710
	COMMON/LYP/NPLY, HUMPLY, ANG, IPLY	00022720
	NP=NPLY(1)+NPLY(2)	00022730
C		00022740
C		00022750

C	FAIL INCREMENTS THE POINT LOAD TO EACH SUCCESSIVE	00022760
C	PLY AND INTERFACE FAILURE UNTIL FINAL JOINT FAILURE	00022770
C	TAKES PLACE	00022780
C		00022790
C	FULL BEARING FAILURE ANALYSIS	00022800
	IF(BPR.NE.0.0) GO TO 600	00022810
	IROUT=1	00022820
C		00022830
C	LOOP OVER ALL PLIES TO FIND LOAD, LOCATION, AND MODE OF	00022840
C	NEXT PLY FAILURE	00022850
		00022860
	100 IF(DELP.EQ.0.) GO TO 10	00022870
	PPF=1.0D10	00022880
	GO TO 15	00022890
	10 PPF=1000.	00022900
	15 MODE=0	00022910
	DELPF=0.00	00022920
	NN=NPLY(1)+NPLY(2)	00022930
	DO 20 I=1,NN	00022940
		00022950
	IF PLY HAS ALREADY LOST STIFFNESS, GO	00022960
	ON TO THE NEXT PLY	00022970
		00022980
	IF(MDAMP(I).EQ.10) GO TO 20	00022990
		00023000
	DETERMINE WHICH PLATE THIS PLY IN IN	00023010
		00023020
	K=1	00023030
	IF(I.GT.NPLY(1)) K=2	00023040
		00023050
	CALCULATE THE LOAD ON PLY FOR CURRENT JOINT LOAD	00023060
		00023070
	PL=-H(K)*(PLYK(I)*U(I)+BARK(I)*BARU(I))	00023080
		00023090
	ASSUME FAILURE OCCURS ONLY ON BEARING SIDE	00023100
		00023110
		00023120
		00023130
		00023140
	IF(PL.LT.0..AND.K.EQ.1) GO TO 20	00023150
	IF(PL.GT.0..AND.K.EQ.2) GO TO 20	00023160
		00023170
	DETERMINE PLY LOAD NECESSARY TO CAUSE NEXT	00023180
	FAILURE AND ITS MODE	00023190
		00023200
	IF(NOPT4.NE.1.AND.NOPT4.NE.3) GO TO 200	00023210
	MODE=8	00023220
	IN=I-(K-1)*NPLY(1)	00023230
	NPY=IPLY(IN,K)	00023240
	PF=PFL(NPY,K)	00023250
		00023260
		00023270
	IF PL>PF AT CURRENT JOINT LOAD	00023280
	PREDICT FAILURE	00023290
		00023300
	NCC=0	00023310
	IF(DELP.NE.0.) GO TO 210	00023320
	IF(DABS(PL).LT.DABS(PF)) GO TO 210	00023330
	PPF=0.	00023340
	INPLY=I	00023350

	MODEF=MODE	00023360
	NCC=1	00023370
	GO TO 140	00023380
200	NMN=1	00023390
	IF(I.GT.NPLY(1)) NMN=I-NPLY(1)	00023400
	NX=IPLY(NMN,K)	00023410
	IF(PBR(NX,K).LT.PSO(NX,K).OR.PNS(NX,K).LT.PSO(NX,K)) GO TO 700	00023420
	MODE=1	00023430
	PF=PSO(NX,K)	00023440
	IF(MDAMP(I).EQ.1) MODE=5	00023450
	IF(MDAMP(I).EQ.1) PF=PALT(1,K)*PF	00023460
	GO TO 25	00023470
700	IF(PNS(NX,K).LT.PBR(NX,K)) GO TO 710	00023480
	MODE=2	00023490
	PF=PBR(NX,K)	00023500
	IF(MDAMP(I).EQ.2) MODE=6	00023510
	IF(MDAMP(I).EQ.2) PF=PALT(2,K)*PF	00023520
	GO TO 25	00023530
710	MODE=3	00023540
	PF=PNS(NX,K)	00023550
	IF(MDAMP(I).EQ.3) MODE=7	00023560
	IF(MDAMP(I).EQ.3) PF=PALT(3,K)*PF	00023570
25	CONTINUE	00023580
	NCC=0	00023590
	IF(DELPH.EQ.0) GO TO 210	00023600
	IF(DABS(PL).LT.DABS(PF)) GO TO 210	00023610
	PFP=0.	00023620
	IIMPLY=1	00023630
	DEF=MODE	00023640
	NCC=1	00023650
	GO TO 1212	00023660
210	CONTINUE	00023670
C		00023680
C	DETERMINE INCREMENTAL JOINT LOAD TO CAUSE	00023690
C	PLY FAILURE	00023700
	IF(ITT.LE.1) GO TO 21	00023710
	IF(DABS(DABS(U(I)/UN(I))-1.).LT.1.0D-10) GO TO 20	00023720
21	CONTINUE	00023730
	DELPH=(PF-DABS(PN(I)))*1000./((DABS(PL)-DABS(PN(I)))	00023740
C		00023750
C	A NEGATIVE VALUE OF DELPH INDICATES UNLOADING	00023760
C	IN A PLY. THIS NODE IS THEN SKIPPED	00023770
C		00023780
C	IF(DELPH.LT.0.) GO TO 20	00023790
C		00023800
C	RECORD LOWEST JOINT FAILURE LOAD INCREMENT,	00023810
C	PLY IN WHICH IT OCCURS, AND MODE	00023820
	PFP2=PFP	00023830
	IF(DELPH.EQ.0) PFP2=1.	00023840
	IF(DELPH.GT.PFP2) GO TO 20	00023850
	PFP=DELPH	00023860
	IIMPLY=1	00023870
	MODEF=MODE	00023880
20	CONTINUE	00023890
C		00023900
C	LOOP OVER ALL INTERFACES TO FIND LOAD	00023910
C	AND LOCATION OF NEXT DELAMINATION	00023920
C		00023930
		00023940
		00023950

	NN=NPLY(1),NPLY(2)-2	00023960
	DO 30 J=1,NN	00023970
C	IF INTERFACE HAS ALREADY FAILED, GO TO NEXT	00023980
C	IF(MDAMI(J).EQ.1) GO TO 50	00023990
		00024000
C	DETERMINE WHICH PLATE INTERFACE IS IN	00024010
		00024020
C	K=1	00024030
	IF(J.GE.NPLY(1)) K=2	00024040
		00024050
C	CALCULATE INTERFACE SHEAR STRAIN FOR CURRENT	00024060
	JOINT LOAD	00024070
C		00024080
	GAMJ=(U(J+K-1)-U(J+K))/H(K)	00024090
C		00024100
	DETERMINE INCREMENTAL JOINT LOAD TO CAUSE	00024110
C	INTERFACE FAILURE	00024120
		00024130
C	IF(ITT.EQ.1) GO TO 47	00024140
	IF(DABS(DABS(GAMJ/GAMN(J))-1.),LT.1.0E-10) GO TO 50	00024150
	CONTINUE	00024160
47	DELPF=(GAMD(K)-DABS(GAMN(J)))/(DABS(GAMJ)-DABS(GAMN(J)))*1000.	00024170
	IF(DELPF.LT.0.) GO TO 50	00024180
		00024190
C		00024200
	RECORD LOWEST JOINT FAILURE LOAD INCREMENT,	00024210
C	PLY OF INTERFACE IN WHICH IT OCCURS, AND MODE	00024220
		00024230
C		00024240
	PFP2=PFP	00024250
	IF(DELP.EQ.0) PFP2=1.	00024260
	IF(DELPF.GT.PFP2) GO TO 50	00024270
	PFP=DELPF	00024280
	INPLY=J	00024290
	MODEF=9	00024300
50	CONTINUE	00024310
		00024320
C		00024330
	DETERMINE VALUES AT END OF INCREMENT	00024340
C		00024350
	JOINT LOAD AT FAILURE	00024360
C		00024370
	IF(MODEF.EQ.0) GO TO 325	00024380
	P=P+PFP*DELP/1000.	00024390
325	CONTINUE	00024400
		00024410
C	NODAL DISPLACEMENTS AND PLY LOADS	00024420
		00024430
C	NN=NPLY(1)+NPLY(2)	00024440
	DO 55 I=1,NN	00024450
	UN(I)=UN(I)+(U(I)-UN(I))*PFP/1000.	00024460
C		00024470
	UPDATE UN	00024480
C		00024490
	IF(NCC.EQ.1) UN(I)=U(I)	00024500
	K=1	00024510
	IF(I.GT.NPLY(1)) K=2	00024520
	PN(I)=-H(K)*PLYK(I)*UN(I)+BARK(I)*BARU(I)	00024530
55	CONTINUE	00024540
C		00024550

C	INTERFACE SHEAR STRAINS	00024560
C		00024570
	NH=NPLY(1)+NPLY(2)-2	00024580
	DO 60 J=1,NH	00024590
	K=1	00024600
	IF(J.GE.NPLY(1)) K=2	00024610
	GAMN(J)=(UN(J+K-1)-UN(J+K))/H(K)	00024620
60	CONTINUE	00024630
1212	CONTINUE	00024640
C		00024650
C	PLY STIFFNESSES, DAMAGE STATES, AND NEXT LOAD	00024660
C	INCREMENT	00024670
		00024680
	K=1	00024690
	IF(INPLY.GT.NPLY(1)) K=2	00024700
	NMN=INPLY	00024710
	IF(INPLY.GT.NPLY(1)) NMN=INPLY-NPLY(1)	00024720
	NX=INPLY(NMN,K)	00024730
	IF(MODEF.NE.0) GO TO 70	00024740
	DELP=1000.	00024750
	NDAM=1	00024760
	GO TO 65	00024770
70	IF(MODEF.NE.1) GO TO 80	00024780
	IF(INPLY.EQ.1.AND.MDAMI(INPLY).EQ.1) GO TO 75	00024790
	IF(INPLY.EQ.NPLY(1).AND.MDAMI(INPLY-1).EQ.1) GO TO 75	00024800
	IF(INPLY.EQ.(NPLY(1)+1).AND.MDAMI(INPLY-1).EQ.1) GO TO 75	00024810
	IF(INPLY.EQ.(NPLY(1)+NPLY(2)).AND.MDAMI(NPLY(1)+NPLY(2)-1).EQ.1) GO TO 75	00024820
	KK=0	00024830
	IF(INPLY.GT.NPLY(1)) KK=1	00024840
	IF(MDAMI(INPLY-KK-1).EQ.1.AND.MDAMI(INPLY-KK).EQ.1) GO TO 75	00024850
	MDAMP(INPLY)=1	00024860
	TEMPK=PLYK(INPLY)	00024870
	PLYK(INPLY)=DELSO(NX,K)*TEMPK	00024880
	BARK(INPLY)=(1.-DELSO(NX,K))*TEMPK	00024890
	BARU(INPLY)=UN(INPLY)	00024900
	DELP=0.	00024910
	NDAM=2	00024920
	GO TO 65	00024930
75	PLYK(INPLY)=0.0	00024940
	BARK(INPLY)=0.	00024950
	BARU(INPLY)=UN(INPLY)	00024960
	MDAMP(INPLY)=10	00024970
	DELP=0.	00024980
	NDAM=2	00024990
	GO TO 65	00025000
80	IF(MODEF.NE.2) GO TO 85	00025010
	TEMPK=PLYK(INPLY)	00025020
	PLYK(INPLY)=DELS(NX,K)*TEMPK	00025030
	BARK(INPLY)=(1.-DELS(NX,K))*TEMPK	00025040
	BARU(INPLY)=UN(INPLY)	00025050
	MDAMP(INPLY)=2	00025060
	DELP=0.	00025070
	NDAM=4	00025080
	GO TO 65	00025090
85	IF(MODEF.NE.3) GO TO 90	00025100
	TEMPK=PLYK(INPLY)	00025110
	PLYK(INPLY)=DELSR(NX,K)*TEMPK	00025120
	BARK(INPLY)=(1.-DELSR(NX,K))*TEMPK	00025130
	BARU(INPLY)=UN(INPLY)	00025140
		00025150

MDAMP(INPLY)=3	00025160
DELP=0.	00025170
NDAM=5	00025180
GO TO 65	00025190
90 IF(MODEF.NE.4) GO TO 140	00025200
PLYK(INPLY)=0.0	00025210
BARK(INPLY)=0.	00025220
BARU(INPLY)=UN(INPLY)	00025230
MDAMP(INPLY)=10	00025240
DELP=0.	00025250
NDAM=6	00025260
GO TO 65	00025270
140 IF(MODEF.NE.5) GO TO 110	00025280
PLYK(INPLY)=0.	00025290
BARK(INPLY)=0.	00025300
BARU(INPLY)=UN(INPLY)	00025310
MDAMP(INPLY)=10	00025320
DELP=0.	00025330
NDAM=6	00025340
GO TO 65	00025350
110 IF(MODEF.NE.6) GO TO 115	00025360
PLYK(INPLY)=0.0	00025370
BARK(INPLY)=0.0	00025380
BARU(INPLY)=UN(INPLY)	00025390
MDAMP(INPLY)=10	00025400
NDAM=7	00025410
DELP=0.	00025420
GO TO 65	00025430
115 IF(MODEF.NE.7) GO TO 120	00025440
PLYK(INPLY)=0.	00025450
BARK(INPLY)=0.	00025460
BARU(INPLY)=UN(INPLY)	00025470
MDAMP(INPLY)=10	00025480
DELP=0.	00025490
NDAM=8	00025500
GO TO 65	00025510
120 IF(MODEF.NE.8) GO TO 125	00025520
PLYK(INPLY)=0.	00025530
BARK(INPLY)=0.	00025540
BARU(INPLY)=UN(INPLY)	00025550
MDAMP(INPLY)=10	00025560
NDAM=9	00025570
DELP=0.	00025580
GO TO 65	00025590
125 IF(MODEF.NE.9) GO TO 65	00025600
DELP=1000.	00025610
MDAMI(INPLY)=1	00025620
NDAM=10	00025630
GO TO 65	00025640
600 CONTINUE	00025650
PARTIAL BEARING FAILURE ANALYSIS	00025660
IROUT=2	00025670
NPL=NPLY(1)+NPLY(2)	00025680
AJFLNS=1.0D10	00025690
AJFLBR=1.0D10	00025700
AJFLSD=1.0D10	00025710
DO 550 I=1,NPL	00025720
K=1	00025730
	00025740
	00025750

IF(I.GT.NPLY(1)) K=2	00025760
PL=-H(K)*PLY(K)*H(K)+BARK(I)*BARU(I)	00025770
IF(PL.LT.0.0.AND.K.EQ.1) GO TO 550	00025780
IF(PL.LT.0.0.AND.K.EQ.2) GO TO 550	00025790
PL=DABS(PL)*ASTM*H(K)*NPLY(K)/1000.	00025800
IJ=1	00025810
IF(I.GT.NPLY(1)) IJ=IJ-NPLY(1)	00025820
IP=IPLY(IJ,K)	00025830
NT=1	00025840
IF(BPSTS(K,IP,2,1).LT.0.0) NT=2	00025850
F1=DABS(PSTC(NT,IP,K)/(BPSTS(K,IP,1,1)+PLMBPSTS(K,IP,2,1)))	00025860
F2=DABS(PSTC(3,IP,K)/(BPSTS(K,IP,1,2)+PLMBPSTS(K,IP,2,2)))	00025870
F3=DABS(PSTC(4,IP,K)/(BPSTS(K,IP,1,3)+PLMBPSTS(K,IP,2,3)))	00025880
IF(MDAMP(I).EQ.5) F1=PALT(3,K)*F1	00025890
IF(MDAMP(I).EQ.4) F2=PALT(2,K)*F2	00025900
IF(MDAMP(I).EQ.2) F3=PALT(1,K)*F3	00025910
IF(AJFLNS.OT.F1) NF1=1	00025920
IF(AJFLNS.OT.F1) AJFLNS=F1	00025930
IF(AJFLBR.OT.F2) NF2=1	00025940
IF(AJFLBR.OT.F2) AJFLBR=F2	00025950
IF(AJFLSO.OT.F3) NF3=1	00025960
IF(AJFLSO.OT.F3) AJFLSO=F3	00025970
550 CONTINUE	00025980
IF(AJFLNS.OT.AJFLBR.OR.AJFLNS.OT.AJFLSO) GO TO 560	00025990
INPLY=NF1	00026000
551 NDAM=5	00026010
IF(MDAMP(INPLY).EQ.2) GO TO 571	00026020
IF(MDAMP(INPLY).EQ.4) GO TO 561	00026030
IF(MDAMP(INPLY).EQ.5) NDAM=8	00026040
MDAMP(INPLY)=NDAM	00026050
PFAIL=AJFLNS	00026060
GO TO 64	00026070
560 IF(AJFLBR.OT.AJFLNS.OR.AJFLBR.OT.AJFLSO) GO TO 571	00026080
INPLY=NF2	00026090
561 NDAM=4	00026100
IF(MDAMP(INPLY).EQ.5) GO TO 551	00026110
IF(MDAMP(INPLY).EQ.2) GO TO 571	00026120
IF(MDAMP(INPLY).EQ.4) NDAM=7	00026130
MDAMP(INPLY)=NDAM	00026140
PFAIL=AJFLBR	00026150
GO TO 64	00026160
570 IF(AJFLSO.OT.AJFLNS.OR.AJFLSO.OT.AJFLBR) GO TO 64	00026170
INPLY=NF3	00026180
571 NDAM=2	00026190
IF(MDAMP(INPLY).EQ.5) GO TO 551	00026200
IF(MDAMP(INPLY).EQ.4) GO TO 561	00026210
IF(MDAMP(INPLY).EQ.2) NDAM=6	00026220
MDAMP(INPLY)=NDAM	00026230
PFAIL=AJFLSO	00026240
64 CONTINUE	00026250
K=1	00026260
IF(INPLY.OT.NPLY(1)) K=2	00026270
IPL=INPLY	00026280
IF(IPL.OT.NPLY(1)) IPL=IPL-NPLY(1)	00026290
IPLP=IPLY(IPL,K)	00026300
ANGLE=ANG(IPLP,K)	00026310
NODE=NPHM(IPL,K)	00026320
IF(MDAMP(INPLY).EQ.6) GO TO 107	00026330
IF(MDAMP(INPLY).EQ.5) AR=DELNS(IPLP,K)	00026340
IF(MDAMP(INPLY).EQ.4) AR=DELBK(IPLP,K)	00026350

	IF(MDAMP(INPLY).EQ.2) AR=DELSO(IPLP,K)	00026360
	TEMPK=PLYK(INPLY)	00026370
	PLYK(INPLY)=AR*TEMPK	00026380
	BARK(INPLY)=(1.-AR)*TEMPK	00026390
	BARU(INPLY)=U(INPLY)	00026400
	ITYP=IPLY(IPL,K)	00026410
	NTFL=0	00026420
	GO TO 103	00026430
107	CONTINUE	00026440
	IF(K.EQ.1) NPLY(1)=NPLY(1)-1	00026450
	IF(K.EQ.2) NPLY(2)=NPLY(2)-1	00026460
	NP=INPLY	00026470
	IF(K.EQ.2) NP=INPLY-NPLY(1)	00026480
	N=NPLY(K)-NP+2	00026490
	ITYP=IPLY(NP,K)	00026500
	DO 101 I=1,N	00026510
	IPLY(NP+I-1,K)=IPLY(NP+I,K)	00026520
	NPNM(NP+I-1,K)=NPNM(NP+I,K)	00026530
101	CONTINUE	00026540
	N=NPL-INPLY	00026550
	DO 102 I=1,N	00026560
	MDAMP(INPLY+I-1)=MDAMP(INPLY+I)	00026570
	PLYK(INPLY+I-1)=PLYK(INPLY+I)	00026580
	BARK(INPLY+I-1)=BARK(INPLY+I)	00026590
102	BARU(INPLY+I-1)=BARU(INPLY+I)	00026600
	NTFL=1	00026610
	NULTF=NULTF-1	00026620
	IF(NULTF.EQ.0) JNT=0	00026630
	IF(NPLY(1).EQ.2.OR.NPLY(2).EQ.2) JNT=0	00026640
103	CONTINUE	00026650
	RETURN	00026660
65	CONTINUE	00026670
C		00026680
C	INCREMENT LOAD IF JOINT HAS NOT FAILED	00026690
	T1=0.	00026700
	T2=0.	00026710
	N1=NPLY(1)	00026720
	N2=NPLY(2)	00026730
	DO 135 I=1,N1	00026740
135	T1=T1+PLYK(I)	00026750
	DO 126 I=1,N2	00026760
	N3=NPLY(1)+I	00026770
126	T2=T2+PLYK(N3)	00026780
	IF(T1.EQ.0.0.OR.T2.EQ.0.0) GO TO 130	00026790
	RETURN	00026800
130	JNT=0	00026810
	RETURN	00026820
	END	00026830
C		00026840
C		00026850
C		00026860
	SUBROUTINE PRINT(U,P,DELP,PFail,ANGLE,BPR,NODE,IROUT,JNT,	00026870
	MNP,MSDLS,ITY)	00026880
		00026890
C		00026900
	IMPLICIT REAL*8(A-H,O-Z)	00026910
	DIMENSION U(100),PLYK(100)	00026920
	DIMENSION NPLY(2),NUNPLY(2),ANG(5,2),IPLY(100,2)	00026930
	DIMENSION BARK(100),BARU(100)	00026940
		00026950

```

DIMENSION NPNM(100.2)
COMMON/COUNT/NPNM
COMMON/LYP/NPLY,NUMPLY,ANG,IPLY
COMMON/PBB/PLYK,DARK,BARU
COMMON/PRT/NDAM,INPLY,ITYP

PRINT VALUES AT END OF INCREMENT

IF(ITT.EQ.1) WRITE(6,10)
10 FORMAT(//,10X,'FAILURE MODE ABBREVIATIONS'
*10X,'ND' = NO ADDITIONAL DAMAGE AT CURRENT JOINT LOAD'
*10X,'DL' = DELAMINATION
*10X,'SO' = SHEAR-OUT
*10X,'BR' = BEARING
*10X,'NS' = NET SECTION
*10X,'SUD' = ULTIMATE FAILURE AFTER SO AND DL
*10X,'SU' = ULTIMATE FAILURE IN SO
*10X,'BU' = ULTIMATE FAILURE IN BR
*10X,'NSU' = ULTIMATE FAILURE IN NS
*10X,'ULT' = ULTIMATE FAILURE
*4X,'INCREMENT NO',3X,'JOINT LOAD',5X,'NODE',8X,'PLY TYPE',
*8X,'MODE',/)
PL=P
IF(ROUT.EQ.2) PL=PFAIL
IF(NSDLS.EQ.2) PL=2.*PL
IF(ITT.EQ.1) PFAILP=0.0D0
IF(PFAILP.LT.PL) PFAILP=PL
IF(JNT.EQ.0.AND.PFAILP.EQ.0.0D0) PFAILP=PL
K=1
IF(INPLY.GT.NPLY(1)) K=2
N=IPLY(INPLY,K)
IF(K.EQ.2) N=IPLY((INPLY-NPLY(1)),K)
IF(ROUT.EQ.2) N=ITYP
IF(BPR.EQ.0.0.OR.BPR.EQ.1.0) ANGLE=ANG(N,K)
IF(BPR.EQ.0.0.OR.BPR.EQ.1.0) NODE=INPLY
IF(NDAM.EQ.1) WRITE(6,20) ITT,PL
IF(NDAM.EQ.2) WRITE(6,30) ITT,PL,NODE,ANGLE
IF(NDAM.EQ.3) WRITE(6,40) ITT,PL,NODE,ANGLE
IF(NDAM.EQ.4) WRITE(6,50) ITT,PL,NODE,ANGLE
IF(NDAM.EQ.5) WRITE(6,60) ITT,PL,NODE,ANGLE
IF(NDAM.EQ.6) WRITE(6,70) ITT,PL,NODE,ANGLE
IF(NDAM.EQ.7) WRITE(6,80) ITT,PL,NODE,ANGLE
IF(NDAM.EQ.8) WRITE(6,90) ITT,PL,NODE,ANGLE
IF(NDAM.EQ.9) WRITE(6,100) ITT,PL,NODE,ANGLE
IF(NDAM.EQ.10) WRITE(6,110) ITT,PL,NODE,ANGLE
20 FORMAT(5X,15,10X,D9.3,34X,'ND')
30 FORMAT(5X,15,10X,D9.3,3X,15,5X,F7.3,' DEGREE ',5X,' SO')
40 FORMAT(5X,15,10X,D9.3,3X,15,5X,F7.3,' DEGREE ',5X,' SUD')
50 FORMAT(5X,15,10X,D9.3,3X,15,5X,F7.3,' DEGREE ',5X,' BR')
60 FORMAT(5X,15,10X,D9.3,3X,15,5X,F7.3,' DEGREE ',5X,' NS')
70 FORMAT(5X,15,10X,D9.3,3X,15,5X,F7.3,' DEGREE ',5X,' SU')
80 FORMAT(5X,15,10X,D9.3,3X,15,5X,F7.3,' DEGREE ',5X,' BU')
90 FORMAT(5X,15,10X,D9.3,3X,15,5X,F7.3,' DEGREE ',5X,' NSU')
100 FORMAT(5X,15,10X,D9.3,3X,15,5X,F7.3,' DEGREE ',5X,' ULT')
110 FORMAT(5X,15,10X,D9.3,3X,15,5X,F7.3,' DEGREE ',5X,' DL')
IF(JNT.EQ.0) GO TO 220
GO TO 250
220 CONTINUE
WRITE(6,240) PFAILP
240 FORMAT(//,' THE PREDICTED JOINT FAILURE ',/,

```

```

00026960
00026970
00026980
00026990
00027000
00027010
00027020
00027030
00027040
00027050
00027060
00027070
00027080
00027090
00027100
00027110
00027120
00027130
00027140
00027150
00027160
00027170
00027180
00027190
00027200
00027210
00027220
00027230
00027240
00027250
00027260
00027270
00027280
00027290
00027300
00027310
00027320
00027330
00027340
00027350
00027360
00027370
00027380
00027390
00027400
00027410
00027420
00027430
00027440
00027450
00027460
00027470
00027480
00027490
00027500
00027510
00027520
00027530
00027540
00027550

```


	*' LOAD IS 'D14.7' LBS',//)	00027560
250	CONTINUE	00027570
	RETURN	00027580
	END	00027590
C		00027600
C		00027610
C		00027620
	SUBROUTINE LINV2F (A,N,IA,AINV,IDGT,WKAREA,IER)	00027630
C	DOUBLE PRECISION A(IA,N),AINV(IA,N),WKAREA(1),ZERO,ONE	00027640
	DATA ONE/1.0D0/,ZERO/0.0D0/	00027650
C	FIRST EXECUTABLE STATEMENT	00027660
C	INITIALIZE IER	00027670
	IER=0	00027680
C	SET AINV TO THE N X N	00027690
C	IDENTITY MATRIX	00027700
	DO 10 I = 1,N	00027710
	DO 5 J = 1,N	00027720
	AINV(I,J) = ZERO	00027730
5	CONTINUE	00027740
	AINV(I,I) = ONE	00027750
10	CONTINUE	00027760
C	COMPUTE THE INVERSE OF A	00027770
	CALL LEQ2F (A,N,N,IA,AINV,IDGT,WKAREA,IER)	00027780
	IF (IER.EQ.0) GO TO 9005	00027790
9000	CONTINUE	00027800
	CALL UERTST (IER,6HLINV2F)	00027810
9005	RETURN	00027820
	END	00027830
C		00027840
C		00027850
	SUBROUTINE LEQ2F (A,M,N,IA,B,IDGT,WKAREA,IER)	00027860
C	DIMENSION A(IA,1),B(IA,1),WKAREA(1)	00027870
	DOUBLE PRECISION A,B,WKAREA,D1,D2,WA	00027880
C	FIRST EXECUTABLE STATEMENT	00027890
C	INITIALIZE IER	00027900
	IER=0	00027910
	JER=0	00027920
	J = N*N+1	00027930
	K = J+N	00027940
	MM = K+N	00027950
	KK = 0	00027960
	MM1 = MM-1	00027970
	JJ=1	00027980
	DO 5 L=1,N	00027990
	DO 5 I=1,N	00028000
	WKAREA(JJ)=A(I,L)	00028010
	JJ=JJ+1	00028020
5	CONTINUE	00028030
C	DECOMPOSE A	00028040
	CALL LUDATN (WKAREA,N,N,A,IA,IDGT,D1,D2,WKAREA(J),WKAREA(K),	00028050
	WA,IER)	00028060
	IF (IER.GT.128) GO TO 25	00028070
	IF (IDGT.EQ.0 .OR. IER.NE.0) KK = 1	00028080
	DO 15 I = 1,M	00028090
C	PERFORMS THE ELIMINATION PART OF	00028100
C	AX = B	00028110
	CALL LUELMM (A,IA,N,B(1,I),WKAREA(J),WKAREA(MM))	00028120
C	REFINEMENT OF SOLUTION TO AX = B	00028130
		00028140
		00028150

	IF (KK .NE. 0)	00028160
*	CALL LUREFN (WKAREA,N,N,A,IA,B(1,1),IDOT,WKAREA(J),WKAREA(MM),	00028170
*	WKAREA(K),WKAREA(K),JER).	00028180
	DO 10 II=1,N	00028190
	B(II,1) = WKAREA(MM1+II)	00028200
10	CONTINUE	00028210
	IF (JER.NE.0) GO TO 20	00028220
15	CONTINUE	00028230
	GO TO 25	00028240
20	IER = 131	00028250
25	JJ=1	00028260
	DO 30 J = 1,N	00028270
	DO 30 I = 1,N	00028280
	A(I,J)=WKAREA(JJ)	00028290
	JJ=JJ+1	00028300
30	CONTINUE	00028310
	IF (IER .EQ. 0) GO TO 9005	00028320
9000	CONTINUE	00028330
	CALL UERTST (IER,6HLEQT2F)	00028340
9005	RETURN	00028350
	END	00028360
		00028370
		00028380
		00028390
	SUBROUTINE LUDATF (A,LU,N,IA,IDOT,D1,D2,IPVT,EQUIL,WA,IER)	00028400
		00028410
	DIMENSION A(IA,1),LU(IA,1),IPVT(1),EQUIL(1)	00028420
	DOUBLE PRECISION A,LU,D1,D2,EQUIL,WA,ZERO,ONE,FOUR,SIXTH,SIXTH,	00028430
*	RN,WREL,BIGA,BIG,P,SUM,AI,WI,T,TEST,Q	00028440
	DATA ZERO,ONE,FOUR,SIXTH,SIXTH/0.D0,1.D0,4.D0,	00028450
*	16.D0,.0625D0/	00028460
	FIRST EXECUTABLE STATEMENT	00028470
	INITIALIZATION	00028480
	IER = 0	00028490
	RN = N	00028500
	WREL = ZERO	00028510
	D1 = ONE	00028520
	D2 = ZERO	00028530
	BIGA = ZERO	00028540
	DO 10 I=1,N	00028550
	BIG = ZERO	00028560
	DO 5 J=1,N	00028570
	P = A(I,J)	00028580
	LU(I,J) = P	00028590
	P = DABS(P)	00028600
	IF (P .GT. BIG) BIG = P	00028610
5	CONTINUE	00028620
	IF (BIG .GT. BIGA) BIGA = BIG	00028630
	IF (BIG .EQ. ZERO) GO TO 110	00028640
	EQUIL(I) = ONE/BIG	00028650
10	CONTINUE	00028660
	DO 105 J=1,N	00028670
	JM1 = J-1	00028680
	IF (JM1 .LT. 1) GO TO 40	00028690
	COMPUTE U(I,J), I=1,...,J-1	00028700
	DO 35 I=1,JM1	00028710
	SUM = LU(I,J)	00028720
	IM1 = I-1	00028730
	IF (IDOT .EQ. 0) GO TO 25	00028740
	WITH ACCURACY TEST	00028750

	AI = DABS(SUM)	00028760
	WI = ZERO	00028770
	IF (IM1 .LT. 1) GO TO 20	00028780
	DO 15 K=1,IM1	00028790
	T = LU(I,K)*LU(K,J)	00028800
	SUM = SUM-T	00028810
	WI = WI+DABS(T)	00028820
15	CONTINUE	00028830
	LU(I,J) = SUM	00028840
20	WI = WI+DABS(SUM)	00028850
	IF (AI .EQ. ZERO) AI = BIGA	00028860
	TEST = WI/AI	00028870
	IF (TEST .GT. WREL) WREL = TEST	00028880
	GO TO 35	00028890
C	WITHOUT ACCURACY	00028900
25	IF (IM1 .LT. 1) GO TO 35	00028910
	DO 30 K=1,IM1	00028920
	SUM = SUM-LU(I,K)*LU(K,J)	00028930
30	CONTINUE	00028940
	LU(I,J) = SUM	00028950
35	CONTINUE	00028960
40	P = ZERO	00028970
C	COMPUTE U(J,J) AND L(I,J), I=J+1,...,	00028980
	DO 70 I=J,N	00028990
	SUM = LU(I,J)	00029000
	IF (IDOT .EQ. 0) GO TO 55	00029010
C	WITH ACCURACY TEST	00029020
	AI = DABS(SUM)	00029030
	WI = ZERO	00029040
	IF (JM1 .LT. 1) GO TO 50	00029050
	DO 45 K=1,JM1	00029060
	T = LU(I,K)*LU(K,J)	00029070
	SUM = SUM-T	00029080
	WI = WI+DABS(T)	00029090
45	CONTINUE	00029100
	LU(I,J) = SUM	00029110
50	WI = WI+DABS(SUM)	00029120
	IF (AI .EQ. ZERO) AI = BIGA	00029130
	TEST = WI/AI	00029140
	IF (TEST .GT. WREL) WREL = TEST	00029150
	GO TO 65	00029160
C	WITHOUT ACCURACY TEST	00029170
55	IF (JM1 .LT. 1) GO TO 65	00029180
	DO 60 K=1,JM1	00029190
	SUM = SUM-LU(I,K)*LU(K,J)	00029200
60	CONTINUE	00029210
	LU(I,J) = SUM	00029220
65	Q = EQUIL(I)*DABS(SUM)	00029230
	IF (P .GE. Q) GO TO 70	00029240
	P = Q	00029250
	IMAX = I	00029260
70	CONTINUE	00029270
C	TEST FOR ALGORITHMIC SINGULARITY	00029280
	IF (RN+P .EQ. RN) GO TO 110	00029290
	IF (J .EQ. IMAX) GO TO 80	00029300
C	INTERCHANGE ROWS J AND IMAX	00029310
	D1 = -D1	00029320
	DO 75 K=1,N	00029330
	P = LU(IMAX,K)	00029340
	LU(IMAX,K) = LU(J,K)	00029350

	LU(I,K) = P	00029360
75	CONTINUE	00029370
	EQUIL(IMAX) = EQUIL(J)	00029380
80	IPVT(J) = IMAX	00029390
	D1 = DIMLU(J,J)	00029400
85	IF (DABS(D1) .LE. ONE) GO TO 90	00029410
	D1 = DIMSIXTH	00029420
	D2 = D2+FOUR	00029430
	GO TO 85	00029440
90	IF (DABS(D1) .GE. SIXTH) GO TO 95	00029450
	D1 = DIMSIXTH	00029460
	D2 = D2-FOUR	00029470
	GO TO 90	00029480
95	CONTINUE	00029490
	JPI = J+1	00029500
	IF (JPI .GT. N) GO TO 105	00029510
C		00029520
	P = LU(J,J)	00029530
	DO 100 I=JPI,N	00029540
	LU(I,J) = LU(I,J)/P	00029550
100	CONTINUE	00029560
105	CONTINUE	00029570
C		00029580
	IF (IDOT .EQ. 0) GO TO 9005	00029590
	P = 3MN+3	00029600
	WA = PMWREL	00029610
	IF (WA+10.D0MN(-IDOT) .NE. WA) GO TO 9005	00029620
	IER = 34	00029630
	GO TO 9000	00029640
C		00029650
	ALGORITHMIC SINGULARITY	00029660
110	IER = 129	00029670
	D1 = ZERO	00029680
	D2 = ZERO	00029690
9000	CONTINUE	00029700
C		00029710
	PRINT ERROR	00029720
9005	CALL UERTST(IER,6HLDATF)	00029730
	RETURN	00029740
	END	00029750
C		00029760
	SUBROUTINE LUELMN (A,IA,N,B,APVT,X)	00029770
C		00029780
	DIMENSION A(IA,1),B(1),APVT(1),X(1)	00029790
	DOUBLE PRECISION A,B,X,SUM,APVT	00029800
C		00029810
	FIRST EXECUTABLE STATEMENT	00029820
C		00029830
	SOLVE LY = B FOR Y	00029840
	DO 5 I=1,N	00029850
5	X(I) = B(I)	00029860
	IW = 0	00029870
	DO 20 I=1,N	00029880
	IP = APVT(I)	00029890
	SUM = X(IP)	00029900
	X(IP) = X(I)	00029910
	IF (IW .EQ. 0) GO TO 15	00029920
	IM1 = I-1	00029930
	DO 10 J=IW,IM1	00029940
	SUM = SUM-A(I,J)*X(J)	00029950
10	CONTINUE	
	GO TO 20	
15	IF (SUM .NE. 0.D0) IW = I	

C	20 X(I) = SUM	00029763
	SOLVE UX = Y FOR X	00029970
	DO 30 I=1,N	00029980
	I = N+1-IB	00029990
	IP1 = I+1	00030000
	SUM = X(I)	00030010
	IF (IP1 .GT. N) GO TO 30	00030020
	DO 25 J=IP1,N	00030030
	SUM = SUM-A(I,J)*X(J)	00030040
	25 CONTINUE	00030050
	30 X(I) = SUM/A(I,I)	00030060
	RETURN	00030070
	END	00030080
C		00030090
G		00030100
C	SUBROUTINE LUREFN (A,IA,N,UL,IUL,B,IDOT,APVT,X,RES,DX,IER)	00030110
	DIMENSION A(IA,1),UL(IUL,1),B(1),X(1),RES(1),DX(1)	00030120
	DIMENSION APVT(1)	00030130
	DIMENSION ACCXT(2)	00030140
	DOUBLE PRECISION A,ACCXT,B,UL,X,RES,DX,ZERO,XNORM,DXNORM,APVT	00030150
	DATA ITMAX/75,ZERO/0.00/	00030160
C	FIRST EXECUTABLE STATEMENT	00030170
	IER=0	00030180
	XNORM = ZERO	00030190
	DO 10 I=1,N	00030200
	XNORM = DMAX1(XNORM,DABS(X(I)))	00030210
	10 CONTINUE	00030220
	IF (XNORM .NE. ZERO) GO TO 20	00030230
	IDOT = 50	00030240
	GO TO 9005	00030250
	20 DO 45 ITER=1,ITMAX	00030260
	DO 30 I=1,N	00030270
	ACCXT(1) = 0.000	00030280
	ACCXT(2) = 0.000	00030290
	CALL VXADD(B(I),ACCXT)	00030300
	DO 25 J=1,N	00030310
	CALL VXMUL(-A(I,J),X(J),ACCXT)	00030320
	25 CONTINUE	00030330
	CALL VXSTO(ACCXT,RES(I))	00030340
	30 CONTINUE	00030350
	CALL LUELMN (UL,IUL,N,RES,APVT,DX)	00030360
	DXNORM = ZERO	00030370
	XNORM = ZERO	00030380
	DO 35 I=1,N	00030390
	X(I) = X(I) + DX(I)	00030400
	DXNORM = DMAX1(DXNORM,DABS(DX(I)))	00030410
	XNORM = DMAX1(XNORM,DABS(X(I)))	00030420
	35 CONTINUE	00030430
	IF (ITER .NE. 1) GO TO 40	00030440
	IDOT = 50	00030450
	IF (DXNORM .NE. ZERO) IDOT = -DLOG10(DXNORM/XNORM)	00030460
	40 IF (XNORM+DXNORM .EQ. XNORM) GO TO 9005	00030470
	45 CONTINUE	00030480
	ITERATION DID NOT CONVERGE	00030490
C	IER = 129	00030500
	9000 CONTINUE	00030510
	CALL UERTST(IER,6HLUREFN)	00030520
	9005 RETURN	00030530
	END	00030540
		00030550

C		00030560
C		00030570
C		00030580
	SUBROUTINE UERTST (IER,NAME)	00030590
C		00030600
	INTEGER IER	00030610
	INTEGER NAME(1)	00030620
C		00030630
	SPECIFICATIONS FOR LOCAL VARIABLES	
	INTEGER I,IEQ,IEQDF,IOUNIT,LEVEL,LEVOLD,NAMEQ(4),	00030640
	NAMSET(6),NAMUPK(6),NIN,NMT	00030650
	DATA NAMSET/1HU,1HE,1HR,1HS,1HE,1HT/	00030660
	DATA NAMEQ/6*1H/	00030670
	DATA LEVEL/4/,IEQDF/0/,IEQ/1H=	00030680
C		00030690
C	UNPACK NAME INTO NAMUPK	00030700
	FIRST EXECUTABLE STATEMENT	00030710
C	CALL USPDK (NAME,6,NAMUPK,NMT)	00030720
	GET OUTPUT UNIT NUMBER	00030730
C	CALL UGETIO(1,NIN,IOUNIT)	00030740
	CHECK IER	00030750
	IF (IER.GT.999) GO TO 25	00030760
	IF (IER.LT.-32) GO TO 55	00030770
	IF (IER.LE.128) GO TO 5	00030780
	IF (LEVEL.LT.1) GO TO 30	00030790
C	PRINT TERMINAL MESSAGE	00030800
	IF (IEQDF.EQ.1) WRITE(IOUNIT,35) IER,NAMEQ,IEQ,NAMUPK	00030810
	IF (IEQDF.EQ.0) WRITE(IOUNIT,35) IER,NAMUPK	00030820
	GO TO 30	00030830
5	IF (IER.LE.64) GO TO 10	00030840
	IF (LEVEL.LT.2) GO TO 30	00030850
C	PRINT WARNING WITH FIX MESSAGE	00030860
	IF (IEQDF.EQ.1) WRITE(IOUNIT,40) IER,NAMEQ,IEQ,NAMUPK	00030870
	IF (IEQDF.EQ.0) WRITE(IOUNIT,40) IER,NAMUPK	00030880
	GO TO 30	00030890
10	IF (IER.LE.32) GO TO 15	00030900
C	PRINT WARNING MESSAGE	00030910
	IF (LEVEL.LT.3) GO TO 30	00030920
	IF (IEQDF.EQ.1) WRITE(IOUNIT,45) IER,NAMEQ,IEQ,NAMUPK	00030930
	IF (IEQDF.EQ.0) WRITE(IOUNIT,45) IER,NAMUPK	00030940
	GO TO 30	00030950
15	CONTINUE	00030960
C	CHECK FOR UERSET CALL	00030970
	DO 20 I=1,6	00030980
	IF (NAMUPK(I).NE.NAMSET(I)) GO TO 25	00030990
20	CONTINUE	00031000
	LEVOLD = LEVEL	00031010
	LEVEL = IER	00031020
	IER = LEVOLD	00031030
	IF (LEVEL.LT.0) LEVEL = 4	00031040
	IF (LEVEL.GT.4) LEVEL = 4	00031050
	GO TO 30	00031060
25	CONTINUE	00031070
	IF (LEVEL.LT.4) GO TO 30	00031080
C	PRINT NON-DEFINED MESSAGE	00031090
	IF (IEQDF.EQ.1) WRITE(IOUNIT,50) IER,NAMEQ,IEQ,NAMUPK	00031100
	IF (IEQDF.EQ.0) WRITE(IOUNIT,50) IER,NAMUPK	00031110
30	IEQDF = 0	00031120
	RETURN	00031130
35	FORMAT(19H *** TERMINAL ERROR,10X,7H(IER = ,13,	00031140
1	20H) FROM IMSL ROUTINE ,6A1,A1,6A1)	00031150
40	FORMAT(27H *** WARNING WITH FIX ERROR,2X,7H(IER = ,13,	

	1	20H) FROM IMSL ROUTINE ,6A1,A1,6A1)	00031160
	45	FORMAT(18H *** WARNING ERROR,11X,7H(IER = ,IS,	00031170
	1	20H) FROM IMSL ROUTINE ,6A1,A1,6A1)	00031180
	50	FORMAT(20H *** UNDEFINED ERROR,9X,7H(IER = ,IS,	00031190
	1	20H) FROM IMSL ROUTINE ,6A1,A1,6A1)	00031200
C			00031210
C		SAVE P FOR P = R CASE	00031220
		P IS THE PAGE NAMUPK	00031230
		R IS THE ROUTINE NAMUPK	00031240
C	55	IEQDF = 1	00031250
		DO 60 I=1,6	00031260
	60	NAMEQ(I) = NAMUPK(I)	00031270
	65	RETURN	00031280
		END	00031290
C			00031300
C			00031310
C		SUBROUTINE UGETID(IOPT,HIN,NOUT)	00031320
		SPECIFICATIONS FOR ARGUMENTS	00031330
C	INTEGER	IOPT,HIN,NOUT	00031340
C		SPECIFICATIONS FOR LOCAL VARIABLES	00031350
	INTEGER	NIND,NOUTD	00031360
C	DATA	NIND/5/,NOUTD/6/	00031370
		FIRST EXECUTABLE STATEMENT	00031380
C		IF (IOPT.EQ.3) GO TO 10	00031390
		IF (IOPT.EQ.2) GO TO 5	00031400
		IF (IOPT.NE.1) GO TO 9005	00031410
		NIN = NIND	00031420
		NOUT = NOUTD	00031430
		GO TO 9005	00031440
	5	NIND = NIN	00031450
		GO TO 9005	00031460
	10	NOUTD = NOUT	00031470
	9005	RETURN	00031480
		END	00031490
C			00031500
C			00031510
C		SUBROUTINE VXADD(A,ACC)	00031520
C			00031530
C		SPECIFICATIONS FOR ARGUMENTS	00031540
	DOUBLE PRECISION	A,ACC(2)	00031550
C		SPECIFICATIONS FOR LOCAL VARIABLES	00031560
	DOUBLE PRECISION	X,Y,Z,ZZ	00031570
C		FIRST EXECUTABLE STATEMENT	00031580
		X = ACC(1)	00031590
		Y = A	00031600
		IF (DABS(ACC(1)).GE.DABS(A)) GO TO 1	00031610
		X = A	00031620
		Y = ACC(1)	00031630
C		COMPUTE Z+ZZ = ACC(1)+A EXACTLY	00031640
	1	Z = X+Y	00031650
		ZZ = (X-Z)+Y	00031660
C		COMPUTE ZZ+ACC(2) USING DOUBLE	00031670
C		PRECISION ARITHMETIC	00031680
		ZZ = ZZ+ACC(2)	00031690
C		COMPUTE ACC(1)+ACC(2) = Z+ZZ EXACTLY	00031700
		ACC(1) = Z+ZZ	00031710
		ACC(2) = (Z-ACC(1))+ZZ	00031720
		RETURN	00031730
			00031740
			00031750

C	END		00031760
C			00031770
C			00031780
C	SUBROUTINE VXMUL (A,B,ACC)		00031790
C		SPECIFICATIONS FOR ARGUMENTS	00031800
C	DOUBLE PRECISION A,B,ACC(2)		00031810
C		SPECIFICATIONS FOR LOCAL VARIABLES	00031820
C	DOUBLE PRECISION X,HA,TA,HB,TB		00031830
C	INTEGER IX(2),I		00031840
C	LOGICAL M1	LX(8),LI(4)	00031850
C	EQUIVALENCE	(X,LX(1),IX(1)),(I,LI(1))	00031860
C	DATA	I/O	00031870
C		SPLIT A = HA+TA	00031880
C		B = HB+TB	00031890
C		FIRST EXECUTABLE STATEMENT	00031900
C	X = A		00031910
C	LI(4) = LX(5)		00031920
C	IX(2) = 0		00031930
C	I = (I/16)*16		00031940
C	LX(5) = LI(4)		00031950
C	HA=X		00031960
C	TA=A-HA		00031970
C	X = B		00031980
C	LI(4) = LX(5)		00031990
C	IX(2) = 0		00032000
C	I = (I/16)*16		00032010
C	LX(5) = LI(4)		00032020
C	HB = X		00032030
C	TB = B-HB		00032040
C		COMPUTE HAHB,HANTB,TANHb, AND TANTB	00032050
C		AND CALL VXADD TO ACCUMULATE THE	00032060
C		SUM	00032070
C	X = TANTB		00032080
C	CALL VXADD(X,ACC)		00032090
C	X = HANTB		00032100
C	CALL VXADD(X,ACC)		00032110
C	X = TANHb		00032120
C	CALL VXADD(X,ACC)		00032130
C	X = HANHb		00032140
C	CALL VXADD(X,ACC)		00032150
C	RETURN		00032160
C	END		00032170
C			00032180
C			00032190
C			00032200
C			00032210
C	SUBROUTINE VXSTO (ACC,D)		00032220
C		SPECIFICATIONS FOR ARGUMENTS	00032230
C	DOUBLE PRECISION ACC(2),D		00032240
C		FIRST EXECUTABLE STATEMENT	00032250
C	D = ACC(1)+ACC(2)		00032260
C	RETURN		00032270
C	END		00032280
C			00032290
C			00032300
C			00032310
C	SUBROUTINE ZRPOLY (A,NDEO,Z,IER)		00032320
C		SPECIFICATIONS FOR ARGUMENTS	00032330
C	INTEGER NDEO,IER		00032340
C	DOUBLE PRECISION A(1),Z(1)		00032350

C		SPECIFICATIONS FOR LOCAL VARIABLES	00032360
	INTEGER	N, NN, J, JJ, I, NM1, ICNT, N2, L, N2, NPI	00032370
	REAL	ETA, RMRE, RINFP, REPSR, RADIX, RLO, XX, YY, SINR,	00032380
	1 REAL	COSR, RMAX, RMIN, X, SC, XM, FT, DX, DF, BND, XXX, ARE	00032390
	DOUBLE PRECISION	PT(101)	00032400
	1 DOUBLE PRECISION	TEMP(101), P(101), QP(101), RK(101), GK(101),	00032410
	2	SVK(101)	00032420
	1 DOUBLE PRECISION	SR, SI, U, V, RA, RB, C, D, A1, A2, A3,	00032430
	2	A6, A7, E, F, G, H, SZR, SZI, RLZR, RLZI,	00032440
	LOGICAL	T, AA, BU, CC, FACTOR, REPSR1, ZERO, ONE, FN	00032450
	COMMON /ZRPQLJ/	ZEROK	00032460
	1	P, QP, RK, QK, SVK, SR, SI, U, V, RA, RB, C, D, A1, A2, A3, A6,	00032470
		A7, E, F, G, H, SZR, SZI, RLZR, RLZI, ETA, ARE, RMRE, N, NN	00032480
		THE FOLLOWING STATEMENTS SET MACHINE	00032490
		CONSTANTS USED IN VARIOUS PARTS OF	00032500
		THE PROGRAM. THE MEANING OF THE	00032510
		FOUR CONSTANTS ARE - REPSR1 THE	00032520
		MAXIMUM RELATIVE REPRESENTATION	00032530
		ERROR WHICH CAN BE DESCRIBED AS	00032540
		THE SMALLEST POSITIVE FLOATING	00032550
		POINT NUMBER SUCH THAT $1 + REPSR1$ IS	00032560
		GREATER THAN 1	00032570
		RINFP THE LARGEST FLOATING-POINT	00032580
		NUMBER	00032590
		REPSR THE SMALLEST POSITIVE	00032600
		FLOATING-POINT NUMBER IF THE	00032610
		EXPONENT RANGE DIFFERS IN SINGLE	00032620
		AND DOUBLE PRECISION THEN REPSR	00032630
		AND RINFP SHOULD INDICATE THE	00032640
		SMALLER RANGE	00032650
		RADIX THE BASE OF THE FLOATING-POINT	00032660
		NUMBER SYSTEM USED	00032670
	DATA	RINFP/27FFFFFFFF/	00032680
	DATA	REPSR/200100000/	00032690
	DATA	RADIX/16.0/	00032700
	DATA	REPSR1/234100000000000000/	00032710
	DATA	ZERO/0.000/ ONE/1.000/	00032720
		ZRPOLY USES SINGLE PRECISION	00032730
		CALCULATIONS FOR SCALING, BOUNDS	00032740
		AND ERROR CALCULATIONS.	00032750
		FIRST EXECUTABLE STATEMENT	00032760
	IER = 0		00032770
	IF (NDEG .GT. 100 .OR. NDEG .LT. 1) GO TO 165		00032780
	ETA = REPSR1		00032790
	ARE = ETA		00032800
	RMRE = ETA		00032810
	RLO = REPSR/ETA		00032820
		INITIALIZATION OF CONSTANTS FOR	00032830
		SHIFT ROTATION	00032840
	XX = .7071068		00032850
	YY = -XX		00032860
	SINR = .9975641		00032870
	COSR = -.06975647		00032880
	N = NDEG		00032890
	NN = N+1		00032900
		ALGORITHM FAILS IF THE LEADING	00032910
		COEFFICIENT IS ZERO.	00032920
	IF (A(1).NE.ZERO) GO TO 5		00032930
	IER = 130		00032940
	GO TO 9000		00032950

C		REMOVE THE ZEROS AT THE ORIGIN AND	00032960
C		ANY	00032970
	5	IF (A(NN).NE.ZERO) GO TO 10	00032980
		J = NDEG-N+1	00032990
		JJ = J+NDEG	00033000
		Z(J) = ZERO	00033010
		Z(JJ) = ZERO	00033020
		NN = NN-1	00033030
		N = N-1	00033040
		IF (NN.EQ.1) GO TO 9005	00033050
		GO TO 5	00033060
C		MAKE A COPY OF THE COEFFICIENTS	00033070
	10	DO 15 I=1,NN	00033080
		P(I) = A(I)	00033090
	15	CONTINUE	00033100
C		START THE ALGORITHM FOR ONE ZERO	00033110
	20	IF (N.GT.2) GO TO 30	00033120
		IF (N.LT.1) GO TO 9005	00033130
C		CALCULATE THE FINAL ZERO OR PAIR OF	00033140
		ZEROS	00033150
		IF (N.EQ.2) GO TO 25	00033160
		Z(NDEG) = -P(2)/P(1)	00033170
		Z(NDEG+NDEG) = ZERO	00033180
		GO TO 145	00033190
	25	CALL ZRPQ(I, P(1), P(2), P(3), Z(NDEG-1), Z(NDEG+NDEG-1), Z(NDEG),	00033200
	1	Z(NDEG+NDEG))	00033210
		GO TO 145	00033220
C		FIND LARGEST AND SMALLEST MODULI OF	00033230
		COEFFICIENTS.	00033240
	30	RMAX = 0.	00033250
		RMIN = RINFP	00033260
		DO 35 I=1,NN	00033270
		X = ABS(SNGL(P(I)))	00033280
		IF (X.GT.RMAX) RMAX = X	00033290
		IF (X.NE.0. AND X.LT.RMIN) RMIN = X	00033300
	35	CONTINUE	00033310
C		SCALE IF THERE ARE LARGE OR VERY	00033320
		SMALL COEFFICIENTS COMPUTES A	00033330
		SCALE FACTOR TO MULTIPLY THE	00033340
		COEFFICIENTS OF THE POLYNOMIAL.	00033350
		THE SCALING IS DONE TO AVOID	00033360
		OVERFLOW AND TO AVOID UNDETECTED	00033370
		UNDERFLOW INTERFERING WITH THE	00033380
		CONVERGENCE CRITERION.	00033390
		THE FACTOR IS A POWER OF THE BASE	00033400
		SC = RLO/RMIN	00033410
		IF (SC.GT.1.0) GO TO 40	00033420
		IF (RMAX.LT.10.) GO TO 55	00033430
		IF (SC.EQ.0.) SC = REPSM*RADIX*RADIX	00033440
		GO TO 45	00033450
	40	IF (RINFP/SC.LT.RMAX) GO TO 55	00033460
	45	L = ALOG(SC)/ALOG(RADIX)+.5	00033470
		IF (L.EQ.0) GO TO 55	00033480
		FACTOR = DBLE(RADIX)**L	00033490
		DO 50 I=1,NN	00033500
	50	P(I) = FACTOR*P(I)	00033510
C		COMPUTE LOWER BOUND ON MODULI OF	00033520
		ZEROS.	00033530
	55	DO 60 I=1,NN	00033540
	60	PT(I) = ABS(SNGL(P(I)))	00033550

	PT(NN) = -PT(NN)		00033560
C		COMPUTE UPPER ESTIMATE OF BOUND	00033570
	X = EXP((ALOG(-PT(NN))-ALOG(PT(1)))/N)		00033580
	IF (PT(N).EQ.0.) GO TO 65		00033590
C		IF NEWTON STEP AT THE ORIGIN IS	00033600
C		BETTER, USE IT.	00033610
	XN = -PT(NN)/PT(N)		00033620
	IF (XM.LT.X) X = XM		00033630
C		CHOP THE INTERVAL (0,X) UNTIL FF.LE.	00033640
65	XM = XM.1		00033650
	FF = PT(1)		00033660
	DO 70 I=2,NN		00033670
70	FF = FF*XM+PT(I)		00033680
	IF (FF.LE.0.) GO TO 75		00033690
	X = XM		00033700
	GO TO 65		00033710
75	DX = X		00033720
C		DO NEWTON ITERATION UNTIL X	00033730
C		CONVERGES TO TWO DECIMAL PLACES	00033740
80	IF (ABS(DX/X).LE..005) GO TO 90		00033750
	FF = PT(1)		00033760
	DF = FF		00033770
	DO 85 I=2,N		00033780
	FF = FF*X+PT(I)		00033790
	DF = DF*X+FF		00033800
85	CONTINUE		00033810
	FF = FF*X+PT(NN)		00033820
	DX = FF/DF		00033830
	X = X-DX		00033840
	GO TO 80		00033850
90	BND = X		00033860
C		COMPUTE THE DERIVATIVE AS THE INITIAL	00033870
C		K POLYNOMIAL AND DO 5 STEPS WITH	00033880
C		NO SHIFT	00033890
	NM1 = N-1		00033900
	FN = ONE/N		00033910
	DO 95 I=2,N		00033920
95	RK(I) = (NN-I)*P(I)*FN		00033930
	RK(1) = P(1)		00033940
	AA = P(NN)		00033950
	BB = P(N)		00033960
	ZEROK = RK(N).EQ.ZERO		00033970
	DO 115 JJ=1,5		00033980
	CC = RK(N)		00033990
	IF (ZEROK) GO TO 105		00034000
C		USE SCALED FORM OF RECURRENCE IF	00034010
C		VALUE OF K AT 0 IS NONZERO	00034020
	T = -AA/CC		00034030
	DO 100 I=1,NM1		00034040
	J = NN-I		00034050
	RK(J) = T*RK(J-1)+P(J)		00034060
100	CONTINUE		00034070
	RK(1) = P(1)		00034080
	ZEROK = DABS(RK(N)).LE.DABS(BB)*ETAN10.		00034090
	GO TO 115		00034100
C		USE UNSCALED FORM OF RECURRENCE	00034110
105	DO 110 I=1,NM1		00034120
	J = NN-I		00034130
	RK(J) = RK(J-1)		00034140
110	CONTINUE		00034150

```

      RK(1) = ZERO
      ZEROK = RK(N).EQ.ZERO
115 CONTINUE
      DO 120 I=1,N
120 TEMP(I) = RK(I)
      DO 140 ICNT=1,20
      XXX = COSRXXX-SINRYYY
      YY = SINRXXX+COSRYYY
      XX = XXX
      SR = BNDXXX
      SI = BNDYYY
      U = -SR-SR
      V = BNDMBND
      CALL ZRPQLB (20*ICNT,NZ)
      IF (NZ.EQ.0) GO TO 130
      J = NDEG-N+1
      JJ = J+NDEG
      Z(J) = SZR
      Z(JJ) = SZI
      NN = NN-NZ
      N = NN-1
      DO 125 I=1,NN
125 P(I) = QP(I)
      IF (NZ.EQ.1) GO TO 20
      Z(J+1) = RLZR
      Z(JJ+1) = RLZI
      GO TO 20
      DO 135 I=1,N
135 RX(I) = TEMP(I)
140 CONTINUE
      IER = 131
      DO 145 I=1,NDEG
      NPI = NDEG+I
      P(I) = Z(NPI)
150 CONTINUE
      NZ = NDEG+NDEG
      J = NDEG

```

```

00034160
00034170
00034180
00034190
00034200
00034210
00034220
00034230
00034240
00034250
00034260
00034270
00034280
00034290
00034300
00034310
00034320
00034330
00034340
00034350
00034360
00034370
00034380
00034390
00034400
00034410
00034420
00034430
00034440
00034450
00034460
00034470
00034480
00034490
00034500
00034510
00034520
00034530
00034540
00034550
00034560
00034570
00034580
00034590
00034600
00034610
00034620
00034630
00034640
00034650
00034660
00034670
00034680
00034690
00034700
00034710
00034720
00034730
00034740
00034750

```

SAVE K FOR RESTARTS WITH NEW SHIFTS

LOOP TO SELECT THE QUADRATIC
CORRESPONDING TO EACH NEW SHIFT

QUADRATIC CORRESPONDS TO A DOUBLE
SHIFT TO A NON-REAL POINT AND ITS
COMPLEX CONJUGATE. THE POINT HAS
MODULUS BND AND AMPLITUDE ROTATED
BY 94 DEGREES FROM THE PREVIOUS
SHIFT

SECOND STAGE CALCULATION, FIXED
QUADRATIC

THE SECOND STAGE JUMPS DIRECTLY TO
ONE OF THE THIRD STAGE ITERATIONS
AND RETURNS HERE IF SUCCESSFUL.
DEFLATE THE POLYNOMIAL, STORE THE
ZERO OR ZEROS AND RETURN TO THE
MAIN ALGORITHM.

IF THE ITERATION IS UNSUCCESSFUL
ANOTHER QUADRATIC IS CHOSEN AFTER
RESTORING K

RETURN WITH FAILURE IF NO
CONVERGENCE WITH 20 SHIFTS

CONVERT ZEROS (Z) IN COMPLEX FORM

DO 155 I=1,NDEO	00034760
Z(N2-1) = Z(J)	00034770
Z(N2) = P(J)	00034780
N2 = N2-2	00034790
J = J-1	00034800
155 CONTINUE	00034810
IF (IER.EQ. 0) GO TO 9005	00034820
C SET UNFOUND ROOTS TO MACHINE INFINITY	00034830
N2 = 2*(NDEO-NN)+3	00034840
DO 160 I=1,N	00034850
Z(N2) = RINFP	00034860
Z(N2+1) = RINFP	00034870
N2 = N2+2	00034880
160 CONTINUE	00034890
GO TO 9000	00034900
165 IER = 129	00034910
9000 CONTINUE	00034920
CALL UERTST (IER,6HZRPLY)	00034930
9005 RETURN	00034940
END	00034950
C	00034960
C	00034970
C	00034980
SUBROUTINE ZRPQLB (L2,NZ)	00034990
C	00035000
INTEGER L2,NZ	00035010
C	00035020
INTEGER N,NN,J,ITYPE,I,IFLAG	00035030
REAL ARE,BETAS,BETAV,ETA,OSS,OTS,OTV,OVV,RMRE,SS,	00035040
TS,TSS,TV,TVV,VV	00035050
1 DOUBLE PRECISION P(101),QP(101),RK(101),QK(101),SVK(101)	00035060
DOUBLE PRECISION SR,SI,U,V,RA,RB,C,D,A1,A2,A3,	00035070
1 A6,A7,E,F,G,H,SZR,SZI,RLZR,RLZI,	00035080
2 SVU,SVV,UI,VI,S,ZERO	00035090
LOGICAL VPASS,SPASS,VTRY,STRY	00035100
COMMON /ZRPQLJ/ P,QP,RK,QK,SVK,SR,SI,U,V,RA,RB,C,D,A1,A2,A3,A6,	00035110
1 A7,E,F,G,H,SZR,SZI,RLZR,RLZI,ETA,ARE,RMRE,N,NN	00035120
DATA ZERO/0.000/	00035130
C	00035140
NZ = 0	00035150
C	00035160
C	00035170
C	00035180
C	00035190
C	00035200
C	00035210
C	00035220
C	00035230
C	00035240
BETAV = .25	00035250
BETAS = .25	00035260
OSS = SR	00035270
OVV = V	00035280
C	00035290
C	00035300
CALL ZRPQLH (NN,U,V,P,QP,RA,RB)	00035310
CALL ZRPQL (ITYPE)	00035320
DO 40 J=1,L2	00035330
C	00035340
C	00035350
CALCULATE NEXT K POLYNOMIAL AND ESTIMATE V	

	CALL ZRPQLF (ITYPE)	00035360
	CALL ZRPQLE (ITYPE)	00035370
	CALL ZRPQLG (ITYPE,UI,VI)	00035380
	VV = VI	00035390
C	ESTIMATE S	00035400
	SS = 0.	00035410
	IF (RK(N).NE.ZERO) SS = -P(NN)/RK(N)	00035420
	TV = 1.	00035430
	TS = 1.	00035440
	IF (J.EQ.1.OR.ITYPE.EQ.3) GO TO 35	00035450
C	COMPUTE RELATIVE MEASURES OF	00035460
C	CONVERGENCE OF S AND V SEQUENCES	00035470
	IF (VV.NE.0.) TV = ABS((VV-OVV)/VV)	00035480
	IF (SS.NE.0.) TS = ABS((SS-OSS)/SS)	00035490
C	IF DECREASING, MULTIPLY TWO MOST	00035500
C	RECENT CONVERGENCE MEASURES	00035510
	TVV = 1.	00035520
	IF (TV.LT.OTV) TVV = TV*OTV	00035530
	TSS = 1.	00035540
	IF (TS.LT.OTS) TSS = TS*OTS	00035550
C	COMPARE WITH CONVERGENCE CRITERIA	00035560
	VPASS = TVV.LT.BETAV	00035570
	SPASS = TSS.LT.BETAS	00035580
	IF (.NOT.(SPASS.OR.VPASS)) GO TO 35	00035590
C	AT LEAST ONE SEQUENCE HAS PASSED THE	00035600
C	CONVERGENCE TEST. STORE VARIABLES	00035610
C	BEFORE ITERATING	00035620
	SVU = U	00035630
	SVV = V	00035640
	DO 5 I=1,N	00035650
5	SVK(I) = RK(I)	00035660
	S = SS	00035670
C	CHOOSE ITERATION ACCORDING TO THE	00035680
C	FASTEST CONVERGING SEQUENCE	00035690
	VTRY = .FALSE.	00035700
	STRY = .FALSE.	00035710
	IF (SPASS AND ((.NOT.VPASS).OR.TSS.LT.TVV)) GO TO 20	00035720
10	CALL ZRPQLC (UI,VI,NZ)	00035730
	IF (NZ.GT.0) RETURN	00035740
C	QUADRATIC ITERATION HAS FAILED. FLAG	00035750
C	THAT IT HAS BEEN TRIED AND	00035760
C	DECREASE THE CONVERGENCE	00035770
C	CRITERION.	00035780
	VTRY = TRUE.	00035790
	BETAU = BETAU*.25	00035800
C	TRY LINEAR ITERATION IF IT HAS NOT	00035810
C	BEEN TRIED AND THE S SEQUENCE IS	00035820
C	CONVERGING	00035830
	IF (STRY.OR.(.NOT.SPASS)) GO TO 25	00035840
	DO 15 I=1,N	00035850
15	RK(I) = SVK(I)	00035860
20	CALL ZRPQLD (S,NZ,IFLAG)	00035870
	IF (NZ.GT.0) RETURN	00035880
C	LINEAR ITERATION HAS FAILED. FLAG	00035890
C	THAT IT HAS BEEN TRIED AND	00035900
C	DECREASE THE CONVERGENCE CRITERION	00035910
	STRY = .TRUE.	00035920
	BETAS = BETAS*.25	00035930
	IF (IFLAG.EQ.0) GO TO 25	00035940
C	IF LINEAR ITERATION SIGNALS AN	00035950

C		ALMOST DOUBLE REAL ZERO ATTEM.	00035960
C		QUADRATIC INTERATION	00035970
	UI = -(S+S)		00035980
	VI = SMS		00035990
	GO TO 10		00036000
C		RESTORE VARIABLES	00036010
25	U = SVU		00036020
	V = SVV		00036030
	DO 30 I=1,N		00036040
30	RK(I) = K(I)		00036050
C		TRY QUADRATIC ITERATION IF IT HAS	00036060
C		NOT BEEN TRIED AND THE V SEQUENCE	00036070
C		IS CONVERGING	00036080
C	IF (VPASS.AND.(.NOT.VTRY)) GO TO 10		00036090
C		RECOMPUTE QP AND SCALAR VALUES TO	00036100
		CONTINUE THE SECOND STAGE	00036110
	CALL ZRPQLH (NN,U,V,P,QP,RA,RB)		00036120
	CALL ZRPQLE (ITYPE)		00036130
35	OVV = VV		00036140
	OSS = SS		00036150
	OTV = TV		00036160
	OTS = TS		00036170
40	CONTINUE		00036180
	RETURN		00036190
	END		00036200
C			00036210
C			00036220
C			00036230
C	SUBROUTINE ZRPQLC (UU,VV,NZ)	SPECIFICATIONS FOR ARGUMENTS	00036240
	INTEGER NZ		00036250
	DOUBLE PRECISION UU,VV		00036260
C		SPECIFICATIONS FOR LOCAL VARIABLES	00036270
	INTEGER N,NN,J,I,ITYPE		00036280
	REAL ARE,EE,ETA,OMP,RELSTP,RMP,RMRH,T,ZM		00036290
	DOUBLE PRECISION P(101),QP(101),RK(101),QK(101),SVK(101)		00036300
	DOUBLE PRECISION SR,SI,U,V,RA,RB,C,D,A1,A2,A3,		00036310
1	A6,A7,E,F,G,H,SZR,SZI,RLZR,RLZI,		00036320
2	UI,VI,ZERO,PT01,ONE		00036330
	LOGICAL TRIED		00036340
	COMMON /ZRPQLJ/		00036350
1	DATA P,QP,RK,QK,SVK,SR,SI,U,V,RA,RB,C,D,A1,A2,A3,A6,		00036360
	A7,E,F,G,H,SZR,SZI,PLZR,RLZI,ETA,ARE,RMRE,N,NN		00036370
	ZERO,PT01,ONE/0.000,0.0100,1.000/		00036380
C		FIRST EXECUTABLE STATEMENT	00036390
	NZ = 0		00036400
C		VARIABLE-SHIFT K-POLYNOMIAL	00036410
C		ITERATION FOR A QUADRATIC FACTOR	00036420
C		CONVERGES ONLY IF THE ZEROS ARE	00036430
C		EQUIMODULAR OR NEARLY SO	00036440
C		UU,VV - COEFFICIENTS OF STARTING	00036450
C		QUADRATIC	00036460
C		NZ - NUMBER OF ZERO FOUND	00036470
	TRIED = .FALSE.		00036480
	U = UU		00036490
	V = VV		00036500
	J = 0		00036510
C		MAIN LOOP	00036520
C	5 CALL ZRPQLI (ONE,U,V,SZR,SZI,RLZR,RLZI)		00036530
C		RETURN IF ROOTS OF THE QUADRATIC ARE	00036540
		REAL AND NOT CLOSE TO MULTIPLE OR	00036550

C		NEARLY EQUAL AND OF OPPOSITE SIGN	00036560
C	IF (DABS(DABS(SZR)-DABS(RLZR)).GT.PT01*DABS(RLZR)) RETURN		00036570
C		EVALUATE POLYNOMIAL BY QUADRATIC	00036580
C		SYNTHETIC DIVISION	00036590
	CALL ZRPQLH (NN,U,V,P,QP,RA,RB)		00036600
C	RMP = DABS(RA-SZR*RB)+DABS(SZI*RB)		00036610
C		COMPUTE A RIGOROUS BOUND ON THE	00036620
C		ROUNDING ERROR IN EVALUTING P	00036630
	ZM = SQRT(ABS(SNGL(V)))		00036640
	EE = 2.*ABS(SNGL(QP(1)))		00036650
	T = -SZR*RB		00036660
	DO 10 I=2,N		00036670
	10 EE = EE*ZM+ABS(SNGL(QP(I)))		00036680
	EE = EE*ZM+ABS(SNGL(RA)+T)		00036690
	EE = (5.*RMRE+4.*ARE)*EE-(5.*RMRE+2.*ARE)*(ABS(SNGL(RA)+T)+		00036700
	1 ABS(SNGL(RB))*ZM)+2.*ARE*ABS(T)		00036710
C		ITERATION HAS CONVERGED SUFFICIENTLY	00036720
C		IF THE POLYNOMIAL VALUE IS LESS	00036730
C		THAN 20 TIMES THIS BOUND	00036740
	IF (RMP.GT.20.*EE) GO TO 15		00036750
	NZ = 2		00036760
	RETURN		00036770
	15 J = J+1		00036780
C		STOP ITERATION AFTER 20 STEPS	00036790
	IF (J.GT.20) RETURN		00036800
	IF (J.LT.2) GO TO 25		00036810
	IF (RELSTP.GT..01.OR.RMP.LT.OMP.OR.TRIED) GO TO 25		00036820
C		A CLUSTER APPEARS TO BE STALLING THE	00036830
C		CONVERGENCE. FIVE FIXED SHIFT	00036840
C		STEPS ARE TAKEN WITH A U,V CLOSE	00036850
C		TO THE CLUSTER	00036860
	IF (RELSTP.LT.ETA) RELSTP = ETA		00036870
	RELSTP = SQRT(RELSTP)		00036880
	U = U-U*RELSTP		00036890
	V = V+V*RELSTP		00036900
	CALL ZRPQLH (NN,U,V,P,QP,RA,RB)		00036910
	DO 20 I=1,5		00036920
	CALL ZRPQLE (ITYPE)		00036930
	CALL ZRPQLF (ITYPE)		00036940
	20 CONTINUE		00036950
	TRIED = .TRUE.		00036960
	J = 0		00036970
	25 OMP = RMP		00036980
C		CALCULATE NEXT K POLYNOMIAL AND NEW	00036990
C		U AND V	00037000
	CALL ZRPQLE (ITYPE)		00037010
	CALL ZRPQLF (ITYPE)		00037020
	CALL ZRPQLE (ITYPE)		00037030
	CALL ZRPQLG (ITYPE,UI,VI)		00037040
C		IF VI IS ZERO THE ITERATION IS NOT	00037050
C		CONVERGING	00037060
	IF (VI.EQ.ZERO) RETURN		00037070
	RELSTP = DABS((VI-V)/VI)		00037080
	U = UI		00037090
	V = VI		00037100
	GO TO 5		00037110
	END		00037120
C			00037130
C			00037140
C			00037150

C	SUBROUTINE ZRPQLD (SSS,NZ,IFLAG)	00037160
	INTEGER NZ,IFLAG	00037170
	DOUBLE PRECISION SSS	00037180
C	SPECIFICATIONS FOR LOCAL VARIABLES	00037190
	INTEGER N,NN,J,I	00037200
	REAL ARE,EE,ETA,OMP,PMR,RMS,RMRE	00037210
	DOUBLE PRECISION P(101),QP(101),RK(101),QK(101),SVK(101)	00037220
	DOUBLE PRECISION SR,SI,U,V,RA,RB,C,D,A1,A2,A3,	00037230
1	A6,A7,E,F,G,H,SR,SZI,RLZR,RLZI,	00037240
2	PV,RKV,T,S,ZERO,PT001	00037250
1	COMMON /ZRPQLJ/	00037260
1	DATA	00037270
	P,QP,RK,QK,SVK,SR,SI,U,V,RA,RB,C,D,A1,A2,A3,A6,	00037280
	A7,E,F,G,H,SR,SZI,RLZR,RLZI,ETA,ARE,RMRE,N,NN	00037290
	ZERO/0.000/,PT001/0.00100/	00037300
C	VARIABLE-SHIFT H POLYNOMIAL	00037310
C	ITERATION FOR A REAL ZERO SSS -	00037320
C	STARTING ITERATE	00037330
C	NZ - NUMBER OF ZERO FOUND	00037340
C	IFLAG - FLAG TO INDICATE A PAIR OF	00037350
C	ZEROS NEAR REAL AXIS	00037360
	FIRST EXECUTABLE STATEMENT	00037370
	NZ = 0	00037380
	S = SSS	00037390
	IFLAG = 0	00037400
	J = 0	00037410
C	MAIN LOOP	00037420
C	5 PV = P(1)	00037430
	EVALUATE P AT S	00037440
	QP(1) = PV	00037450
	DO 10 I=2,NN	00037460
	PV = PVMS+P(I)	00037470
	QP(I) = PV	00037480
10	CONTINUE	00037490
	RMP = DABS(PV)	00037500
C	COMPUTE A RIGOROUS BOUND ON THE	00037510
C	ERROR IN EVALUATING P	00037520
	RMS = DABS(S)	00037530
	EE = (RMRE/(ARE+RMRE))*ABS(SNGL(QP(1)))	00037540
	DO 15 I=2,NN	00037550
15	EE = EE*RMS+ABS(SNGL(QP(I)))	00037560
C	ITERATION HAS CONVERGED SUFFICIENTLY	00037570
C	IF THE POLYNOMIAL VALUE IS LESS	00037580
C	THAN 20 TIMES THIS BOUND	00037590
	IF (RMP.GT.20.*((ARE+RMRE)*EE-RMRE*RMP)) GO TO 20	00037600
	NZ = 1	00037610
	SZR = S	00037620
	SZI = ZERO	00037630
	RETURN	00037640
20	J = J+1	00037650
C	STOP ITERATION AFTER 10 STEPS	00037660
	IF (J.GT.10) RETURN	00037670
	IF (J.LT.2) GO TO 25	00037680
	IF (DABS(T).GT.PT001*DABS(S-T).OR.RMP.LE.OMP) GO TO 25	00037690
C	A CLUSTER OF ZEROS NEAR THE REAL	00037700
C	AXIS HAS BEEN ENCOUNTERED RETURN	00037710
C	WITH IFLAG SET TO INITIATE A	00037720
C	QUADRATIC ITERATION	00037730
	IFLAG = 1	00037740
	SSS = S	00037750
	RETURN	

C		RETURN IF THE POLYNOMIAL VALUE HAS	00037760
C		INCREASED SIGNIFICANTLY	00037770
C	25 OMP = RMP		00037780
C		COMPUTE T, THE NEXT POLYNOMIAL, AND	00037790
		THE NEW ITERATE	00037800
	RKV = RK(1)		00037810
	QK(1) = RKV		00037820
	DO 30 I=2,N		00037830
	RKV = RKVMS+RK(I)		00037840
	QK(I) = RKV		00037850
	50 CONTINUE		00037860
	IF (DABS(RKV).LE.DABS(RK(N))*10.*ETA) GO TO 40		00037870
C		USE THE SCALED FORM OF THE	00037880
C		RECURRENCE IF THE VALUE OF K AT S	00037890
C		IS NONZERO	00037900
	T = -PV/RKV		00037910
	RK(1) = QP(1)		00037920
	DO 35 I=2,N		00037930
	35 RK(I) = T*QK(I-1)+QP(I)		00037940
	GO TO 50		00037950
C		USE UNSCALED FORM	00037960
	40 RK(1) = ZERO		00037970
	DO 45 I=2,N		00037980
	45 RK(I) = QK(I-1)		00037990
	50 RKV = RK(1)		00038000
	DO 55 I=2,N		00038010
	55 RKV = RKVMS+RK(I)		00038020
	T = ZERO		00038030
	IF (DABS(RKV).GT.DABS(RK(N))*10.*ETA) T = -PV/RKV		00038040
	S = S+T		00038050
	GO TO 5		00038060
	END		00038070
C			00038080
C			00038090
C			00038100
C	IMSL ROUTINE NAME - ZRPQLE		00038110
C			00038120
C			00038130
C	COMPUTER - IBM/DOUBLE		00038140
C			00038150
C	LATEST REVISION - JANUARY 1, 1978		00038160
C			00038170
C			00038180
C	SUBROUTINE ZRPQLE (ITYPE)		00038190
C		SPECIFICATIONS FOR ARGUMENTS	00038200
C	INTEGER ITYPE		00038210
C		SPECIFICATIONS FOR LOCAL VARIABLES	00038220
	INTEGER N,NN		00038230
	REAL ARE,ETA,RMRE		00038240
	DOUBLE PRECISION P(101),QP(101),RK(101),QK(101),SVK(101)		00038250
	DOUBLE PRECISION SR,S1,U,V,RA,RB,C,D,A1,A2,A3,		00038260
	A6,A7,E,F,G,H,SZR,SZI,RLZR,RLZI		00038270
	1 COMMON /ZRPQLJ/ P,QP,RK,QK,SVK,SR,S1,U,V,RA,RB,C,D,A1,A2,A3,A6,		00038280
	1 A7,E,F,G,H,SZR,SZI,RLZR,RLZI,ETA,ARE,RMRE,N,NN		00038290
C		THIS ROUTINE CALCULATES SCALAR	00038300
C		QUANTITIES USED TO COMPUTE THE	00038310
C		NEXT K POLYNOMIAL AND NEW	00038320
C		ESTIMATES OF THE QUADRATIC	00038330
C		COEFFICIENTS	00038340
C		ITYPE - INTEGER VARIABLE SET HERE	00038350

```

C          INDICATING HOW THE CALCULATIONS ARE NORMALIZED TO AVOID OVERFLOW
C          SYNTHETIC DIVISION OF K BY THE QUADRATIC 1,U,V
C          FIRST EXECUTABLE STATEMENT
C          CALL ZRPQLH (N,U,V,RK,QK,C,D)
C          IF (DABS(C).GT.DABS(RK(N))*100.*ETA) GO TO 5
C          IF (DABS(D).GT.DABS(RK(N-1))*100.*ETA) GO TO 5
C          ITYPE = 3
C          TYPE=3 INDICATES THE QUADRATIC IS ALMOST A FACTOR OF K
C          RETURN
C          5 IF (DABS(D).LT.DABS(C)) GO TO 10
C          ITYPE = 2
C          TYPE=2 INDICATES THAT ALL FORMULAS ARE DIVIDED BY D
C          E = RA/D
C          F = C/D
C          G = U*RB
C          H = V*RB
C          A3 = (RA+G)*E+H*(RB/D)
C          A1 = RB*F-RA
C          A7 = (F+U)*RA+H
C          RETURN
C          10 ITYPE = 1
C          TYPE=1 INDICATES THAT ALL FORMULAS ARE DIVIDED BY C
C          E = RA/C
C          F = D/C
C          G = U*E
C          H = V*RB
C          A3 = RA*E+(H/C+G)*RB
C          A1 = RB-RA*(D/C)
C          A7 = RA+G*D+H*F
C          RETURN
C          END
C          SUBROUTINE ZRPQLF (ITYPE)
C          INTEGER          ITYPE          SPECIFICATIONS FOR ARGUMENTS
C          INTEGER          N,NN,I          SPECIFICATIONS FOR LOCAL VARIABLES
C          REAL             ARE,ETA,RMRE
C          DOUBLE PRECISION P(101),QP(101),RK(101),QK(101),SVK(101)
C          DOUBLE PRECISION SR,SI,U,V,RA,RB,C,D,A1,A2,A3,
C          1               A6,A7,E,F,G,H,SZR,SZI,RLZR,RLZI,TEMP,ZERO
C          COMMON /ZRPQLJ/ P,QP,RK,QK,SVK,SR,SI,U,V,RA,RB,C,D,A1,A2,A3,A6,
C          1               A7,E,F,G,H,SZR,SZI,RLZR,RLZI,ETA,ARE,RMRE,N,NN
C          DATA             ZERO/0.000/
C          COMPUTES THE NEXT K POLYNOMIALS
C          USING SCALARS COMPUTED IN ZRPQLE
C          FIRST EXECUTABLE STATEMENT
C          IF (ITYPE.EQ.3) GO TO 20
C          TEMP = RA
C          IF (ITYPE.EQ.1) TEMP = RB
C          IF (DABS(A1).GT.DABS(TEMP)*ETA*10.) GO TO 10
C          IF A1 IS NEARLY ZERO THEN USE A
C          SPECIAL FORM OF THE RECURRENCE

```

```

00038360
00038370
00038380
00038390
00038400
00038410
00038420
00038430
00038440
00038450
00038460
00038470
00038480
00038490
00038500
00038510
00038520
00038530
00038540
00038550
00038560
00038570
00038580
00038590
00038600
00038610
00038620
00038630
00038640
00038650
00038660
00038670
00038680
00038690
00038700
00038710
00038720
00038730
00038740
00038750
00038760
00038770
00038780
00038790
00038800
00038810
00038820
00038830
00038840
00038850
00038860
00038870
00038880
00038890
00038900
00038910
00038920
00038930
00038940
00038950

```


C	10	B1 = -RK(N)/P(NN)	00039560
		B2 = -(RK(N-1)+B1*P(N))/P(NN)	00039570
		C1 = VMB2*A1	00039580
		C2 = B1*A7	00039590
		C3 = B1*B1*A3	00039600
		C4 = C1-C2-C3	00039610
		TEMP = A5+B1*A4-C4	00039620
		IF (TEMP.EQ.ZERO) GO TO 15	00039630
		UU = U-(U*(C3+C2)+V*(B1*A1+B2*A7))/TEMP	00039640
		VV = V*(1+C4/TEMP)	00039650
		RETURN	00039660
C		IF TYPE=3 THE QUADRATIC IS ZEROED	00039670
	15	UU = ZERO	00039680
		VV = ZERO	00039690
		RETURN	00039700
		END	00039710
C			00039720
C			00039730
C			00039740
C		SUBROUTINE ZRPOLH (NN,U,V,P,Q,RA,RB)	00039750
		SPECIFICATIONS FOR ARGUMENTS	00039760
C		INTEGER NN	00039770
		DOUBLE PRECISION P(NN),Q(NN),U,V,RA,RB	00039780
C		SPECIFICATIONS FOR LOCAL VARIABLES	00039790
		INTEGER I	00039800
		DOUBLE PRECISION C	00039810
C		DIVIDES P BY THE QUADRATIC 1,U,V	00039820
C		PLACING THE QUOTIENT IN Q AND THE	00039830
C		REMAINDER IN A,B	00039840
C		FIRST EXECUTABLE STATEMENT	00039850
		RB = P(1)	00039860
		Q(1) = RB	00039870
		RA = P(2)-U*RB	00039880
		Q(2) = RA	00039890
		DO 5 I=3,NN	00039900
		C = P(I)-U*RA-V*RB	00039910
		Q(I) = C	00039920
		RB = RA	00039930
		RA = C	00039940
	5	CONTINUE	00039950
		RETURN	00039960
		END	00039970
C			00039980
C			00039990
C			00040000
C			00040010
C		IMSL ROUTINE NAME - ZRPQLI	00040020
C			00040030
C			00040040
C			00040050
C		COMPUTER - IBM/DOUBLE	00040060
C			00040070
C		LATEST REVISION - JANUARY 1, 1978	00040080
C			00040090
C		SUBROUTINE ZRPQLI (RA,B1,C,SR,SI,RLR,RLI)	00040100
		SPECIFICATIONS FOR ARGUMENTS	00040110
		DOUBLE PRECISION RA,B1,C,SR,SI,RLR,RLI	00040120
C		SPECIFICATIONS FOR LOCAL VARIABLES	00040130
		DOUBLE PRECISION RE,D,E,ZERO,ONE,TWO	00040140
		DATA ZERO,ONE,TWO/0.000,1.000,2.000/	00040150

```
C
CCCCCCCCC
      CALCULATE THE ZEROS OF THE QUADRATIC
      ANZM**2 + B1XZ + C. THE QUADRATIC
      FORMULA, MODIFIED TO AVOID
      OVERFLOW, IS USED TO FIND THE
      LARGER ZERO IF THE ZEROS ARE REAL
      AND BOTH ZEROS ARE COMPLEX.
      THE SMALLER REAL ZERO IS FOUND
      DIRECTLY FROM THE PRODUCT OF THE
      ZEROS C/A
      FIRST EXECUTABLE STATEMENT
      00040160
      00040170
      00040180
      00040190
      00040200
      00040210
      00040220
      00040230
      00040240
      00040250
      00040260
      00040270
      00040280
      00040290
      00040300
      00040310
      00040320
      00040330
      00040340
      00040350
      00040360
      00040370
      00040380
      00040390
      00040400
      00040410
      00040420
      00040430
      00040440
      00040450
      00040460
      00040470
      00040480
      00040490
      00040500
      00040510
      00040520
      00040530
      00040540
      00040550
      00040560
      00040570
      00040580
      00040590
      00040600
      00040610
      00040620
      00040630
      00040640
      00040650
      00040660
      00040670
      00040680
      00040690
      00040700
      00040710
      00040720
      00040730
      00040740
      00040750
      00040760
      00040770
      00040780
      00040790
      00040800
      00040810
      00040820
      00040830
      00040840
      00040850
      00040860
      00040870
      00040880
      00040890
      00040900
      00040910
      00040920
      00040930
      00040940
      00040950
      00040960
      00040970
      00040980
      00040990
      00041000
      00041010
      00041020
      00041030
      00041040
      00041050
      00041060
      00041070
      00041080
      00041090
      00041100
      00041110
      00041120
      00041130
      00041140
      00041150
      00041160
      00041170
      00041180
      00041190
      00041200
      00041210
      00041220
      00041230
      00041240
      00041250
      00041260
      00041270
      00041280
      00041290
      00041300
      00041310
      00041320
      00041330
      00041340
      00041350
      00041360
      00041370
      00041380
      00041390
      00041400
      00041410
      00041420
      00041430
      00041440
      00041450
      00041460
      00041470
      00041480
      00041490
      00041500
      00041510
      00041520
      00041530
      00041540
      00041550
      00041560
      00041570
      00041580
      00041590
      00041600
      00041610
      00041620
      00041630
      00041640
      00041650
      00041660
      00041670
      00041680
      00041690
      00041700
      00041710
      00041720
      00041730
      00041740
      00041750
      00041760
      00041770
      00041780
      00041790
      00041800
      00041810
      00041820
      00041830
      00041840
      00041850
      00041860
      00041870
      00041880
      00041890
      00041900
      00041910
      00041920
      00041930
      00041940
      00041950
      00041960
      00041970
      00041980
      00041990
      00042000
      00042010
      00042020
      00042030
      00042040
      00042050
      00042060
      00042070
      00042080
      00042090
      00042100
      00042110
      00042120
      00042130
      00042140
      00042150
      00042160
      00042170
      00042180
      00042190
      00042200
      00042210
      00042220
      00042230
      00042240
      00042250
      00042260
      00042270
      00042280
      00042290
      00042300
      00042310
      00042320
      00042330
      00042340
      00042350
      00042360
      00042370
      00042380
      00042390
      00042400
      00042410
      00042420
      00042430
      00042440
      00042450
      00042460
      00042470
      00042480
      00042490
      00042500
      00042510
      00042520
      00042530
      00042540
      00042550
      00042560
      00042570
      00042580
      00042590
      00042600
      00042610
      00042620
      00042630
      00042640
      00042650
      00042660
      00042670
      00042680
      00042690
      00042700
      00042710
      00042720
      00042730
      00042740
      00042750
      00042760
      00042770
      00042780
      00042790
      00042800
      00042810
      00042820
      00042830
      00042840
      00042850
      00042860
      00042870
      00042880
      00042890
      00042900
      00042910
      00042920
      00042930
      00042940
      00042950
      00042960
      00042970
      00042980
      00042990
      00043000
      00043010
      00043020
      00043030
      00043040
      00043050
      00043060
      00043070
      00043080
      00043090
      00043100
      00043110
      00043120
      00043130
      00043140
      00043150
      00043160
      00043170
      00043180
      00043190
      00043200
      00043210
      00043220
      00043230
      00043240
      00043250
      00043260
      00043270
      00043280
      00043290
      00043300
      00043310
      00043320
      00043330
      00043340
      00043350
      00043360
      00043370
      00043380
      00043390
      00043400
      00043410
      00043420
      00043430
      00043440
      00043450
      00043460
      00043470
      00043480
      00043490
      00043500
      00043510
      00043520
      00043530
      00043540
      00043550
      00043560
      00043570
      00043580
      00043590
      00043600
      00043610
      00043620
      00043630
      00043640
      00043650
      00043660
      00043670
      00043680
      00043690
      00043700
      00043710
      00043720
      00043730
      00043740
      00043750
      00043760
      00043770
```

CALCULATE THE ZEROS OF THE QUADRATIC
 AX^2+BX+C . THE QUADRATIC
 FORMULA, MODIFIED TO AVOID
 OVERFLOW, IS USED TO FIND THE
 LARGER ZERO IF THE ZEROS ARE REAL
 AND BOTH ZEROS ARE COMPLEX.
 THE SMALLER REAL ZERO IS FOUND
 DIRECTLY FROM THE PRODUCT OF THE
 ZEROS C/A
 FIRST EXECUTABLE STATEMENT

```

      IF (RA.NE.ZERO) GO TO 10
      SR = ZERO
      IF (BI.NE.ZERO) SR = -C/BI
      RLR = ZERO
5      SI = ZERO
      RLI = ZERO
      RETURN
10     IF (C.NE.ZERO) GO TO 15
      SR = ZERO
      RLR = -BI/RA
      GO TO 5

```

COMPUTE DISCRIMINANT AVOIDING OVERFLOW

```

15 RB = B1/TWO
   IF (DABS(RB).LT.DABS(C)) GO TO 20
   E = ONE-(RA/RB)*(C/RB)
   D = DSQRT(DABS(E))*DABS(RB)
   GO TO 25
20 E = RA
   IF (C.LT.ZERO) E = -RA
   E = RB*(RB/DABS(C))-E
   D = DSQRT(DABS(E))*DSQRT(DABS(C))
25 IF (E.LT.ZERO) GO TO 30

```

REAL ZEROS

```
IF (RB.GE.ZERO) D = -D
RLR = (-RB+D)/RA
SR = ZERO
IF (RLR.NE.ZERO) SR = (C/RLR)/RA
GO TO 5
```

COMPLEX CONJUGATE ZEROS

```
30 SR = -RB/RA
   RLR = SR
   SI = DABS(D/RA)
   RLI = -SI
   RETURN
   END
```

SUBROUTINE LEQ2C (A,N,IA,B,M,IB,IJOB,WA,WK,IER)

```

COMPLEX*16      A(TA,1),B(TB,1),W(H,1),TEMPA,TEMPC,TEMPC
DOUBLE PRECISION WK(N),TA(2),TB(2),TC(2)
DOUBLE PRECISION AR,AI,BR,BI,CR,CI,DXNORM,XNORM,ZERO
DOUBLE PRECISION ACC(2)
EQUIVALENCE
      (TA(1),TEMPA),(TB(1),TEMPC),(TC(1),TEMPC),
      (TA(1),AR),(TA(2),AI),(TB(1),BR),(TB(2),BI),
      (TC(1),CR),(TC(2),CI)
      ZERO/0.0D0/
      ITMAX/50/
DATA
DATA

```

00040160
00040170
00040180
00040190
00040200
00040210
00040220
00040230
00040240
00040250
00040260
00040270
00040280
00040290
00040300
00040310
00040320
00040330
00040340
00040350
00040360
00040370
00040380
00040390
00040400
00040410
00040420
00040430
00040440
00040450
00040460
00040470
00040480
00040490
00040500
00040510
00040520
00040530
00040540
00040550
00040560
00040570
00040580
00040590
00040600
00040610
00040620
00040630
00040640
00040650
00040660
00040670
00040680
00040690
00040700
00040710
00040720
00040730
00040740
00040750

	FIRST EXECUTABLE STATEMENT	
C	IER = 0	00040760
	N1 = N+1	00040770
	N2 = N+2	00040780
	IF (IJOB .EQ. 2) GO TO 15	00040790
C	SAVE MATRIX A	00040800
	DO 10 I = 1,N	00040810
	DO 5 J = 1,N	00040820
	WA(I,J) = A(I,J)	00040830
5	CONTINUE	00040840
10	CONTINUE	00040850
C	FACTOR MATRIX A	00040860
	CALL LEQ1C(WA,N,N,B,M,1B,1,WK,IER)	00040870
	IF (IER .NE. 0) GO TO 9000	00040880
	IF (IJOB .EQ. 1) GO TO 9005	00040890
C	SAVE THE RIGHT HAND SIDES	00040900
15	DO 65 J = 1,M	00040910
	DO 20 I = 1,N	00040920
	WA(I,N1) = B(I,J)	00040930
20	CONTINUE	00040940
C	OBTAIN A SOLUTION	00040950
	CALL LEQ1C(WA,N,N,WA(1,N1),1,N,2,WK,IER)	00040960
C	COMPUTE THE NORM OF THE SOLUTION	00040970
	XNORM = ZERO	00040980
	DO 25 I = 1,N	00040990
	TEMPA = WA(I,N1)	00041000
	XNORM = DMAX1(XNORM,DABS(AR),DABS(AI))	00041010
25	CONTINUE	00041020
	IF (XNORM .EQ. ZERO) GO TO 65	00041030
C	COMPUTE RESIDUALS	00041040
	DO 50 ITER = 1,ITMAX	00041050
	DO 40 I = 1,N	00041060
	TEMPB = B(I,J)	00041070
	ACC(1) = 0.000	00041080
	ACC(2) = 0.000	00041090
	CALL VXADD(BR,ACC)	00041100
	DO 30 JJ = 1,N	00041110
	TEMPA = A(I,JJ)	00041120
	TEMPB = WA(JJ,N1)	00041130
	CALL VXMUL(-AR,PR,ACC)	00041140
	CALL VXMUL(AI,BI,ACC)	00041150
30	CONTINUE	00041160
	CALL VXSTO(ACC,CR)	00041170
	TEMPB = B(I,J)	00041180
	ACC(1) = 0.000	00041190
	ACC(2) = 0.000	00041200
	CALL VXADD(BI,ACC)	00041210
	DO 35 JJ = 1,N	00041220
	TEMPA = A(I,JJ)	00041230
	TEMPB = WA(JJ,N1)	00041240
	CALL VXMUL(-AR,BI,ACC)	00041250
	CALL VXMUL(-BR,AI,ACC)	00041260
35	CONTINUE	00041270
	CALL VXSTO(ACC,C2)	00041280
	WA(I,N2) = TEMPC	00041290
40	CONTINUE	00041300
	CALL LEQ1C(WA,N,N,WA(1,N2),1,N,2,WK,IER)	00041310
	DXNORM = ZERO	00041320
	UPDATE THE SOLUTION	00041330
	DO 45 I = 1,N	00041340
		00041350

	WA(I,N1) = WA(I,N1)+WA(I,N2)	00041360
	TEMPA = WA(I,N2)	00041370
	DXNORM = DMAX1(DXNORM,DABS(AR),DABS(AI))	00041380
45	CONTINUE	00041390
	IF (XNORM+DXNORM .EQ. XNORM) GO TO 55	00041400
50	CONTINUE	00041410
	IER = 130	00041420
C	STORE THE SOLUTION	00041430
55	DO 60 JK = 1,N	00041440
	B(JK,J) = WA(JK,N1)	00041450
60	CONTINUE	00041460
	IF (IER .NE. 0) GO TO 9000	00041470
65	CONTINUE	00041480
	GO TO 9005	00041490
9000	CONTINUE	00041500
	CALL UERTST(IER,AHLEQ2C)	00041510
9005	RETURN	00041520
	END	00041530
C		00041540
C		00041550
C		00041560
C	SUBROUTINE LEQ1C (A,N,IA,B,M,IB,IJOB,WA,IER)	00041570
	SPECIFICATIONS FOR ARGUMENTS	00041580
	INTEGER N,IA,M,IB,IJOB,IER	00041590
	COMPLEX*16 A(IA,N),B(IB,M)	00041600
	DOUBLE PRECISION WA(N)	00041610
C	SPECIFICATIONS FOR LOCAL VARIABLES	00041620
	DOUBLE PRECISION P,Q,ZERO,ONE,T(2),RN,BIG	00041630
	COMPLEX*16 SUM,TEMP	00041640
	INTEGER I,J,JM1,IM1,K,IMAX,JP1,IN,N1	00041650
	EQUIVALENCE (SUM,T(1))	00041660
	DATA ZERO/0.0D0/,ONE/1.D0/	00041670
C	INITIALIZATION	00041680
C	FIRST EXECUTABLE STATEMENT	00041690
	IER = 0	00041700
	IF (IJOB .EQ. 2) GO TO 75	00041710
	RN = N	00041720
C	FIND EQUILIBRATION FACTORS	00041730
	DO 10 I=1,N	00041740
	BIG = ZERO	00041750
	DO 5 J=1,N	00041760
	TEMP = A(I,J)	00041770
	P = CDABS(TEMP)	00041780
	IF (P .GT. BIG) BIG = P	00041790
5	CONTINUE	00041800
	IF (BIG .EQ. ZERO) GO TO 105	00041810
	WA(I) = ONE/BIG	00041820
10	CONTINUE	00041830
C	L-U DECOMPOSITION	00041840
	DO 70 J = 1,N	00041850
	JM1 = J-1	00041860
	IF (JM1 .LT. 1) GO TO 25	00041870
C	COMPUTE U(I,J), I=1,...,J-1	00041880
	DO 20 I=1,JM1	00041890
	SUM = A(I,J)	00041900
	IM1 = I-1	00041910
	IF (IM1 .LT. 1) GO TO 20	00041920
	DO 15 K=1,IM1	00041930
	SUM = SUM-A(I,K)*A(K,J)	00041940
15	CONTINUE	00041950

	A(I,J) = SUM	00041960
20	CONTINUE	00041970
25	P = ZERO	00041980
C	COMPUTE U(J,J) AND L(I,J), I=J+1,....	00041990
	DO 45 I=J,N	00042000
	SUM = A(I,J)	00042010
	IF (JMI .LT. 1) GO TO 40	00042020
	DO 35 K=1,JMI	00042030
	SUM = SUM-A(I,K)*A(K,J)	00042040
35	CONTINUE	00042050
	A(I,J) = SUM	00042060
40	Q = WA(I)*CDABS(SUM)	00042070
	IF (P .GE. Q) GO TO 45	00042080
	P = Q	00042090
	IMAX = I	00042100
45	CONTINUE	00042110
C	TEST FOR ALGORITHMIC SINGULARITY	00042120
	Q = RN+P	00042130
	IF (Q .EQ. RN) GO TO 105	00042140
	IF (J .EQ. IMAX) GO TO 60	00042150
C	INTERCHANGE ROWS J AND IMAX	00042160
	DO 50 K=1,N	00042170
	TEMP = A(IMAX,K)	00042180
	A(IMAX,K) = A(J,K)	00042190
	A(J,K) = TEMP	00042200
50	CONTINUE	00042210
	WA(IMAX) = WA(J)	00042220
60	WA(J) = IMAX	00042230
	JPI = J+1	00042240
	IF (JPI .GT. N) GO TO 70	00042250
C	DIVIDE BY PIVOT ELEMENT U(J,J)	00042260
	TEMP = A(J,J)	00042270
	DO 65 I = JPI,N	00042280
	A(I,J) = A(I,J)/TEMP	00042290
65	CONTINUE	00042300
70	CONTINUE	00042310
75	IF (IJOB .EQ. 1) GO TO 9005	00042320
	DO 103 K = 1,M	00042330
C	SOLVE UX = Y FOR X	00042340
	IW = 0	00042350
	DO 90 I = 1,N	00042360
	IMAX = WA(I)	00042370
	SUM = B(IMAX,K)	00042380
	B(IMAX,K) = B(I,K)	00042390
	IF (IW .EQ. 0) GO TO 85	00042400
	IM1 = I-1	00042410
	DO 80 J = IW,IM1	00042420
	SUM = SUM-A(I,J)*B(J,K)	00042430
80	CONTINUE	00042440
	GO TO 88	00042450
85	IF (T(1) .NE. ZERO .OR. T(2) .NE. ZERO) IW = I	00042460
88	B(I,K) = SUM	00042470
90	CONTINUE	00042480
C	SOLVE LY = B FOR Y	00042490
	N1 = N+1	00042500
	DO 100 IW = 1,N	00042510
	I = N1-IW	00042520
	JPI = I+1	00042530
	SUM = B(I,K)	00042540
	IF (JPI .GT. N) GO TO 98	00042550

	DO 95 J = JP1,N	00042560
	SUM = SUM-A(I,J)*B(J,K)	00042570
95	CONTINUE	00042580
98	B(I,K) = SUM/A(I,I)	00042590
100	CONTINUE	00042600
103	CONTINUE	00042610
	GO TO 9005	00042620
C		00042630
	105 IER = 129	00042640
	9000 CONTINUE	00042650
C		00042660
	CALL UERTST(IER,6HLEQTIC)	00042670
	9005 RETURN	00042680
	END	00042690

ALGORITHMIC SINGULARITY

PRINT ERROR

APPENDIX B
SAMCJ Program Listing


```

COMMON/RT/R
COMMON/MFS/F5CD
COMMON/NTP/NELTYP
COMMON/NPT/NOPT2,NOPT6,NOPT7,NOPT8
COMMON/MD/E1,E2,G12,V12,V21
COMMON/LYP/NPLY,NUMPLY,ANG,IPLY
COMMON/ELP/AX,8X,NOUT,NSTS
COMMON/FCC/ELWDTH,ELTHK,ELLOAD
COMMON/NCST/NCASE,NTYPE
COMMON/DISP/ANR2
COMMON/PBB/PLYK,BARK,BARU
COMMON/ELS/ELSTFF,ELSTSS
COMMON/CMT2/XOUT,YOUT
COMMON/SER/NT,NB
DATA Y,Y'/
DATA CMC/C'/

```

CCCCC

READ IN REQUIRED INPUT DATA

```

WRITE(6,876)
876 FORMAT(///,10X,' PROGRAM SAMCJ',///,
* PROGRAM SAMCJ PREDICTS THE FAILURE LOAD, FAILURE '///,
* LOCATION, AND FAILURE MODE IN MULTIPLY-FASTENED, '///,
* SINGLE OR DOUBLE LAP COMPOSITE SHEAR JOINTS. '///,
* THE ANALYSIS ASSUMES THAT INPUT PARAMETERS ARE '///,
* SPECIFIED IN ENGLISH UNITS - LENGTH IN INCHES, '///,
* MODULI AND STRENGTHS IN PSI. '///)
WRITE(6,900)
900 FORMAT(' ENTER: '///,
* 1 FOR SLS (SINGLE LAP SHEAR)',///,
* 2 FOR DLS (DOUBLE LAP SHEAR)',///)
READ(5,M) NSDLS
WRITE(6,911)
911 FORMAT(' ENTER: '///,
* 1 FOR STATIC TENSION '///,
* 2 FOR STATIC COMPRESSION',///)
READ(5,M) LTNCM
106 FORMAT(A1)
380 CONTINUE
DO 300 K=1,2
IF(K.EQ.1) WRITE(6,912)
IF(K.EQ.2) WRITE(6,913)
912 FORMAT(' IS THE TOP PLATE A COMPOSITE OR A METAL?')
913 FORMAT(' IS THE BOTTOM PLATE A COMPOSITE OR A METAL?')
WRITE(6,914)
914 FORMAT(' ENTER C OR M IN THE FIRST FIELD')
READ(5,106) CM(K)
WRITE(6,203)
203 FORMAT(' INPUT MATERIAL DESCRIPTION OF THIS PLATE '///,
* EX: ASA/3501-6')
READ(5,204) (MTL(K,I),I=1,15)
204 FORMAT(15A4)
300 CONTINUE
IF(CM(1).NE.CMC.OR.CM(2).NE.CMC) WRITE(6,754)
754 FORMAT(//,' NOTE: FOR COMPUTATIONAL PURPOSES A '///,
* METALLIC PLATE IS MODELED AS A 30 PLY '///,
* LAMINATE OF 0 DEGREE PLYS WITH ISOTROPIC '///,
* MATERJ' PROPERTIES',/)

```

```

00000560
00000570
00000580
00000590
00000600
00000610
00000620
00000630
00000640
00000650
00000660
00000670
00000680
00000690
00000700
00000710
00000720
00000730
00000740
00000750
00000760
00000770
00000780
00000790
00000800
00000810
00000820
00000830
00000840
00000850
00000860
00000870
00000880
00000890
00000900
00000910
00000920
00000930
00000940
00000950
00000960
00000970
00000980
00000990
00001000
00001010
00001020
00001030
00001040
00001050
00001060
00001070
00001080
00001090
00001100
00001110
00001120
00001130
00001140
00001150

```

DO 306 K=1,2	00001160
IF(K.EQ.1) WRITE(6,216)	00001170
IF(K.EQ.2) WRITE(6,555)	00001180
216 FORMAT(' INPUT THE ENGINEERING PROPERTIES OF THE TOP PLATE')	00001190
555 FORMAT(' INPUT THE ENGINEERING PROPERTIES OF THE BOTTOM PLATE')	00001200
IF(CM(K).EQ.CMC) GO TO 85	00001210
WRITE(6,95)	00001220
95 FORMAT(' INPUT YOUNGS MODULUS AND POISSONS RATIO')	00001230
READ(5,M) E1(K),V12(K)	00001240
E2(K)=E1(K)	00001250
G12(K)=E1(K)/(2.0D0*V12(K))	00001260
V21(K)=V12(K)*E2(K)/E1(K)	00001270
GO TO 306	00001280
85 CONTINUE	00001290
WRITE(6,217)	00001300
217 FORMAT(' INPUT YOUNGS MODULI, E1 AND E2')	00001310
READ(5,M) E1(K),E2(K)	00001320
WRITE(6,213)	00001330
213 FORMAT(' INPUT THE SHEAR MODULUS AND MAJOR POISSONS RATIO')	00001340
READ(5,M) G12(K),V12(K)	00001350
V21(K)=V12(K)*E2(K)/E1(K)	00001360
306 CONTINUE	00001370
307 CONTINUE	00001380
290 CONTINUE	00001390
DO 303 K=1,2	00001400
IF(CM(K).EQ.CMC) GO TO 45	00001410
NUMPLY(K)=1	00001420
GO TO 303	00001430
45 CONTINUE	00001440
IF(K.EQ.1) WRITE(6,207)	00001450
IF(K.EQ.2) WRITE(6,702)	00001460
207 FORMAT(' INPUT TOTAL NUMBER OF DISTINCT PLY ',/,	00001470
' ORIENTATIONS IN THE TOP PLATE')	00001480
702 FORMAT(' INPUT TOTAL NUMBER OF DISTINCT PLY ',/,	00001490
' ORIENTATIONS IN THE BOTTOM PLATE')	00001500
READ(5,M) NUMPLY(K)	00001510
303 CONTINUE	00001520
DO 209 K=1,2	00001530
IF(CM(K).EQ.CMC) GO TO 55	00001540
ANG(1,K)=0.	00001550
GO TO 209	00001560
55 CONTINUE	00001570
N=NUMPLY(K)	00001580
DO 209 L=1,N	00001590
WRITE(6,206) L	00001600
206 FORMAT(' INPUT ORIENTATION OF PLY TYPE NO',I5)	00001610
READ(5,M) ANG(L,K)	00001620
209 CONTINUE	00001630
WRITE(6,1823)	00001640
1823 FORMAT(/, ' THICKNESS VARIATIONS MAY BE APPROXIMATED',/,	00001650
' BY ASSIGNING DIFFERENT LAYUPS TO ELEMENTS',/,	00001660
' IN A COMPOSITE PLATE OR BY SPECIFYING DIFFERENT',/,	00001670
' THICKNESSES TO ELEMENTS IN A METALLIC PLATE',/)	00001680
IF(NSDLS.EQ.2) WRITE(6,789)	00001690
789 FORMAT(/, ' NOTE: FOR THE DOUBLE LAP SHEAR CASE, FOR',/,	00001700
' THE BOTTOM PLATE, ENTER ONLY HALF FOR THE',/,	00001710
' LAYUP FOR A COMPOSIT OR HALF THE THICKNESS',/,	00001720
' FOR A METALLIC',/)	00001730
DO 811 I=1,2	00001740
IF(CM(I).EQ.CMC) GO TO 891	00001750

NELPLS(I,1)=30	00001760
DO 892 III=1,30	00001770
892 NELPT(I,1,III)=1	00001780
GO TO 811	00001790
891 CONTINUE	00001800
IF(I.EQ.1) WRITE(6,812)	00001810
IF(I.EQ.2) WRITE(6,813)	00001820
812 FORMAT(/, ' ENTER NUMBER OF DIFFERENT LAYUPS IN THE ',/,	00001830
* ' TOP PLATE')	00001840
813 FORMAT(/, ' ENTER NUMBER OF DIFFERENT LAYUPS IN THE ',/,	00001850
* ' BOTTOM PLATE')	00001860
READ(5,M) NL	00001870
DO 814 J=1,NL	00001880
WRITE(6,815) J	00001890
815 FORMAT(' ENTER NUMBER OF PLIES IN LAYUP NO ',I5)	00001900
READ(5,M) NELPLS(I,J)	00001910
WRITE(6,816)	00001920
816 FORMAT(' ENTER PLY THICKNESS FOR THIS LAYUP')	00001930
READ(5,M) PLYTHK(I,J)	00001940
NN=NELPLS(I,J)	00001950
WRITE(6,818)	00001960
818 FORMAT(' ENTER SEQUENCE OF PLY TYPES FROM TOP TO BOTTOM')	00001970
DO 817 K=1,NN	00001980
READ(5,M) NELPT(I,J,K)	00001990
817 CONTINUE	00020000
814 CONTINUE	00020010
811 CONTINUE	00020020
WRITE(6,855)	00020030
855 FORMAT(/, ' FASTENER DESCRIPTION:',/)	00020040
WRITE(6,250)	00020050
250 FORMAT(' INPUT MATERIAL DESCRIPTION FOR FASTENER')	00020060
READ(5,251) (MTL(3,I),I=1,15)	00020070
251 FORMAT(15A4)	00020080
WRITE(6,252)	00020090
252 FORMAT(' INPUT YOUNG'S MODULUS AND POISSON'S RATIO FOR',/,	00020100
* ' THE FASTENER')	00020110
READ(5,M) FASE,FASV	00020120
WRITE(6,253)	00020130
253 FORMAT(' INPUT THE DIAMETER OF THE FASTENER')	00020140
READ(5,M) FASD	00020150
WRITE(6,888)	00020160
888 FORMAT(/, ' FASTENER TYPE',/,	00020170
* ' ENTER: 1 FOR PROTRUDING HEAD',/,	00020180
* ' 2 FOR COUNTERSUNK HEAD')	00020190
READ(5,M) NFTYP	00020200
R(1)=1.0D10	00020210
R(2)=1.0D10	00020220
IF(NFTYP.EQ.1) GO TO 360	00020230
WRITE(6,889)	00020240
889 FORMAT(/, ' ENTER PLATE WHICH CONTAINS THE COUNTERSUNK',/,	00020250
* ' HEAD (OPPOSITE PLATE ASSUMES THE NUT HEAD)',/,	00020260
* ' ENTER: 1 FOR TOP PLATE',/,	00020270
* ' 2 FOR BOTTOM PLATE ')	00020280
READ(5,M) N	00020290
R(N)=0.0D0	00020300
360 CONTINUE	00020310
WRITE(6,477)	00020320
477 FORMAT(/, ' GRID LAYOUT:',/)	00020330
C INPUT GRIDS, ELEMENT CONNECTIVITY AND PROPERTIES	00020340
	00020350

00002360
00002370
00002380
00002390
00002400
00002410
00002420
00002430
00002440
00002450
00002460
00002470
0000248
00002490
00002500
00002510
00002520
00002530
00002540
00002550
00002560
00002570
00002580
00002590
00002600
00002610
00002620
00002630
00002640
00002650
00002660
00002670
00002680
00002690
00002700
00002710
00002720
00002730
00002740
00002750
00002760
00002775
00002780
00002790
00002800
00002810
00002820
00002830
00002840
00002850
00002860
00002870
00002880
00002890
00002900
00002910
00002920
00002930
00002940
00002950

IF(CM(1).EQ.CMC) GO TO 627	00002960
IF(NELTYP(I).NE.1) GO TO 721	00002970
WRITE(6,1721)	00002980
1721 FORMAT(' ENTER ELEMENT THICKNESS')	00002990
READ(5,X) ATH	00003000
721 IF(NELTYP(I).NE.2) GO TO 722	00003010
WRITE(6,723)	00003020
723 FORMAT(' ENTER ELEMENT THICKNESS')	00003030
READ(5,X) ATH	00003040
FSCD(I,1)=GCOORD(NELCHA(I,6),1)	00003050
FSCD(I,2)=GCOORD(NELCHA(I,6),2)	00003060
FSCD(I,3)=FASD/2.000	00003070
722 IF(NELTYP(I).NE.3) GO TO 724	00003080
WRITE(6,725)	00003090
725 FORMAT(' ENTER ELEMENT THICKNESS, X AND Y COORDINATES',/,	00003100
* OF OPEN HOLE AND HOLE RADIUS')	00003110
READ(5,X) ATH,(FSCD(I,J),J=1,3)	00003120
724 ELTHK(I)=ATH/30.000	00003130
PLYTHK(I,1)=ATH/30.000	00003140
LYPN(I)=1	00003150
GO TO 474	00003160
627 CONTINUE	00003170
IF(NELTYP(I).NE.1) GO TO 726	00003180
WRITE(6,727)	00003190
727 FORMAT(' ENTER ELEMENT LAYUP NO')	00003200
READ(5,X) LYPN(I)	00003210
726 IF(NELTYP(I).NE.2) GO TO 728	00003220
WRITE(6,729)	00003230
729 FORMAT(' ENTER ELEMENT LAYUP NO')	00003240
READ(5,X) LYPN(I)	00003250
FSCD(I,1)=GCOORD(NELCHA(I,6),1)	00003260
FSCD(I,2)=GCOORD(NELCHA(I,6),2)	00003270
FSCD(I,3)=FASD/2.000	00003280
728 IF(NELTYP(I).NE.3) GO TO 730	00003290
WRITE(6,731)	00003300
731 FORMAT(' ENTER ELEMENT LAYUP NUMBER, X AND Y ',/,	00003310
* COORDINATES OF THE OPEN HOLE AND THE HOLE',/,	00003320
* RADIUS')	00003330
READ(5,X) LYPN(I),(FSCD(I,J),J=1,3)	00003340
730 ELTHK(I)=PLYTHK(1,LYPN(I))	00003350
474 CONTINUE	00003360
WRITE(6,688)	00003370
688 FORMAT(/, ' ENTER NUMBER OF ELEMENTS IN BOTTOM PLATE '	00003380
READ(5,X) NEL2	00003390
NELTOT=NEL1+NEL2	00003400
NP1=NEL1+1	00003410
DO 611 I=NP1,NELTOT	00003420
WRITE(6,800) I	00003430
800 FORMAT(' FOR ELEMENT NO',I5,	00003440
* ENTER: ELEMENT ID,N1,N2,N3,N4,N5,ELEMENT TYPE')	00003450
READ(5,X) (NELCON(I,J),J=1,6),NELTYP(I)	00003460
DO 593 IL=2,6	00003470
IC=0	00003480
NELCHA(I,1)=NELCON(I,1)	00003490
NIN=NOPI+1	00003500
DO 594 KL=NIN,NGTOT	00003510
IF(NELCON(I,IL).EQ.NGRID(KL)) IC=1	00003520
IF(NELCON(I,IL).EQ.NGRID(KL)) NELCHA(I,IL)=KL	00003530
IF(IC.EQ.1) GO TO 593	00003540
594 CONTINUE	00003550

593	CONTINUE	00003560
	IF(CM(2).EQ.CMC) GO TO 927	00003570
	IF(NELTYP(I).NE.1) GO TO 921	00003580
	WRITE(6,1921)	00003590
1921	FORMAT(' ENTER ELEMENT THICKNESS')	00003600
	READ(5,M) ATH	00003610
921	IF(NELTYP(I).NE.2) GO TO 922	00003620
	WRITE(6,1923)	00003630
1923	FORMAT(' ENTER ELEMENT THICKNESS')	00003640
	READ(5,M) ATH	00003650
	FSCD(I,1)=GCOORD(NELCNA(I,6),1)	00003660
	FSCD(I,2)=GCOORD(NELCNA(I,6),2)	00003670
	FSCD(I,3)=FASD/2.0D0	00003680
	WRITE(6,3443) I,NELCNA(I,6),FSCD(I,1),FSCD(I,2)	00003690
3443	FORMAT(' C I NELCNA FSCD12',I5,2X,5(D9.3,2X))	00003700
922	IF(NELTYP(I).NE.3) GO TO 924	00003710
	WRITE(6,925)	00003720
925	FORMAT(' ENTER ELEMENT THICKNESS, X AND Y COORDINATES',/,	00003730
	* OF OPEN HOLE AND HOLE RADIUS')	00003740
	READ(5,M) ATH,(FSCD(I,J),J=1,3)	00003750
924	ELTHK(I)=ATH/30.0D0	00003760
	PLYTHK(2,1)=ATH/30.0D0	00003770
	LYPN(I)=1	00003780
	GO TO 611	00003790
927	CONTINUE	00003800
	IF(NELTYP(I).NE.1) GO TO 926	00003810
	WRITE(6,1927)	00003820
1927	FORMAT(' ENTER ELEMENT LAYUP NO')	00003830
	READ(5,M) LYPN(I)	00003840
926	IF(NELTYP(I).NE.2) GO TO 928	00003850
	WRITE(6,929)	00003860
929	FORMAT(' ENTER ELEMENT LAYUP NO')	00003870
	READ(5,M) LYPN(I)	00003880
	FSCD(I,1)=GCOORD(NELCNA(I,6),1)	00003890
	FSCD(I,2)=GCOORD(NELCNA(I,6),2)	00003900
	FSCD(I,3)=FASD/2.0D0	00003910
928	IF(NELTYP(I).NE.3) GO TO 930	00003920
	WRITE(6,931)	00003930
931	FORMAT(' ENTER ELEMENT LAYUP NUMBER, X AND Y ',/,	00003940
	* COORDINATES OF THE OPEN HOLE AND THE HOLE',/,	00003950
	* RADIUS')	00003960
	READ(5,M) LYPN(I),(FSCD(I,J),J=1,3)	00003970
930	ELTHK(I)=PLYTHK(2,LYPN(I))	00003980
611	CONTINUE	00003990
	WRITE(6,1741)	00004000
1741	FORMAT(/,' FASTENERS ARE MODELED BY EFFECTIVE ',/,	00004010
	* FASTENER ELEMENTS WHICH PROVIDE THE ',/,	00004020
	* ELASTIC LINK BETWEEN THE TOP AND ',/,	00004030
	* BOTTOM PLATES',/)	00004040
	WRITE(6,1711)	00004050
1711	FORMAT(' ENTER NUMBER OF FASTENERS IN JOINT '	00004060
	READ(5,M) NUMF	00004070
	WRITE(6,,16)	00004080
716	FORMAT(/,'	00004090
	* EFFECTIVE FASTENER ELEMENTS ARE	00004100
	* NUMBERED AS SHOWN:	00004110
	* N1 (TOP PLATE)	00004120
	* N2 (BOTTOM PLATE)	00004130
		00004140
		00004150

```

X' WHERE N1 AND N2 CORRESPOND TO THE CENTRAL
X' NODES IN LOADED HOLE ELEMENTS
X' FORMAT: ELEMENT ID, N1, N2
DO 717 I=1,NUMF
WRITE(6,711) I
711 FORMAT(' ENTER ELEMENT NO',I5)
READ(5,X) (NELFAS(I,J),J=1,3)
717 CONTINUE

C
C
C DETERMINE GRID STORAGE LOCATIONS FOR
ELEMENT NUDES

DO 612 I=1,NEL1
N=6
IF(NELTYP(I).NE.2) N=5
NELCNA(I,1)=NELCON(I,1)
DO 613 J=2,N
IC=0
DO 614 K=1,NOP1
IF(NELCON(I,J).EQ.NGRID(K)) IC=1
IF(NELCON(I,J).EQ.NGRID(K)) NELCNA(I,J)=K
IF(IC.EQ.1) GO TO 613
614 CONTINUE
613 CONTINUE
612 CONTINUE
NPI=NEL1+1
DO 395 I=NPI,NELTOT
N=6
IF(NELTYP(I).NE.2) N=5
NELCNA(I,1)=NELCON(I,1)
DO 616 J=2,N
IC=0
NIN=NOP1+1
DO 617 K=NIN,NGTOT
IF(NELCON(I,J).EQ.NGRID(K)) IC=1
IF(NELCON(I,J).EQ.NGRID(K)) NELCNA(I,J)=K
IF(IC.EQ.1) GO TO 616
617 CONTINUE
616 CONTINUE
395 CONTINUE
DO 741 J=1,NUMF
N=2
NELFSA(I,1)=NELFAS(I,1)
DO 242 J=1,N
IC=0
DO 243 K=1,NGTOT
IF(NELFAS(I,J+1).EQ.NGRID(K)) IC=1
IF(NELFAS(I,J+1).EQ.NGRID(K)) NELFSA(I,J+1)=K
IF(IC.EQ.1) GO TO 242
243 CONTINUE
242 CONTINUE
741 CONTINUE

C
C
C COMPUTE ELEMENT WIDTHS

DO 239 I=1,NELTOT
ELWIDTH(I)=DABS(GCOORD(NELCNA(I,3),2)-GCOORD(NELCNA(I,2),2))
239 CONTINUE
C

```

```

00004160
00004170
00004180
00004190
00004200
00004210
00004220
00004230
00004240
00004250
00004260
00004270
00004280
00004290
00004300
00004310
00004320
00004330
00004340
00004350
00004360
00004370
00004380
00004390
00004400
00004410
00004420
00004430
00004440
00004450
00004460
00004470
00004480
00004490
00004500
00004510
00004520
00004530
00004540
00004550
00004560
00004570
00004580
00004590
00004600
00004610
00004620
00004630
00004640
00004650
00004660
00004670
00004680
00004690
00004700
00004710
00004720
00004730
00004740
00004750

```

C C C C	GROUP ELEMENTS TO AVOID THE DUPLICATE CALCULATION OF IDENTICAL STIFFNESS MATRICIES WRITE(6,3000) 3000 FORMAT(/,' TO REDUCE RUN TIMES, ELEMENTS MAY BE ',/, *' GROUPED INTO SETS WHICH WILL BE ASSIGNED',/, *' IDENTICAL STIFFNESS MATRICIES',/, *' ENTER: 1 TO USE THIS OPTION ',/, *' 2 OTHERWISE ') READ(5,*) NOPT IF(NOPT.EQ.1) GO TO 3001 N1=0 N2=0 N3=0 DO 3002 I=1,NEL1 IF(HELTP(I).EQ.1) N1=N1+1 IF(HELTP(I).EQ.2) N2=N2+1 3002 IF(HELTP(I).EQ.3) N3=N3+1 NEF(1)=NUMF NLH(1)=N2 NOH(1)=N3 NPL(1)=N1 N=NUMF DO 3003 I=1,N 3003 NOFF(1,1,1)=NELFAS(I,1) IC=0 DO 3004 I=1,NEL1 IF(HELTP(I).EQ.2) IC=IC+1 IF(HELTP(I).EQ.2) NGLH(1,IC,1)=NELCON(I,1) IF(HELTP(I).EQ.2) NUMLH(1,IC)=1 3004 CONTINUE IC=0 DO 3005 I=1,NEL1 IF(HELTP(I).EQ.3) IC=IC+1 IF(HELTP(I).EQ.3) NOOH(1,IC,1)=NELCON(I,1) IF(HELTP(I).EQ.3) NUMOH(1,IC)=1 3005 CONTINUE IC=0 DO 3006 I=1,NEL1 IF(HELTP(I).EQ.1) IC=IC+1 IF(HELTP(I).EQ.1) NGPL(1,IC,1)=NELCON(I,1) IF(HELTP(I).EQ.1) NUMPL(1,IC)=1 3006 CONTINUE N=NEL1+1 N1=0 N2=0 N3=0 DO 3007 I=N,NELTOT IF(HELTP(I).EQ.1) N1=N1+1 IF(HELTP(I).EQ.2) N2=N2+1 3007 IF(HELTP(I).EQ.3) N3=N3+1 NEF(2)=NUMF NLH(2)=N2 NOH(2)=N3 NPL(2)=N1 N=NUMF DO 3008 I=1,N 3008 NGEF(2,1,1)=NELFAS(I,1) IC=0	00004760 00004770 00004780 00004790 00004800 00004810 00004820 00004830 00004840 00004850 00004860 00004870 00004880 00004890 00004900 00004910 00004920 00004930 00004940 00004950 00004960 00004970 00004980 00004990 00005000 00005010 00005020 00005030 00005040 00005050 00005060 00005070 00005080 00005090 00005100 00005110 00005120 00005130 00005140 00005150 00005160 00005170 00005180 00005190 00005200 00005210 00005220 00005230 00005240 00005250 00005260 00005270 00005280 00005290 00005300 00005310 00005320 00005330 00005340 00005350
------------------	--	--

```

N=NEL1+1
DO 3010 I=N,NELTOT
IF(NELTYP(I).EQ.2) IC=IC+1
IF(NELTYP(I).EQ.2) NGLH(2,IC,1)=NELCON(I,1)
IF(NELTYP(I).EQ.2) NUMLH(2,IC)=1
3010 CONTINUE
IC=0
N=NEL1+1
DO 3011 I=N,NELTOT
IF(NELTYP(I).EQ.1) IC=IC+1
IF(NELTYP(I).EQ.1) NGPL(2,IC,1)=NELCON(I,1)
IF(NELTYP(I).EQ.1) NUMPL(2,IC)=1
3011 CONTINUE
IC=0
N=NEL1+1
DO 3012 I=N,NELTOT
IF(NELTYP(I).EQ.3) IC=IC+1
IF(NELTYP(I).EQ.3) NMOH(2,IC,1)=NELCON(I,1)
IF(NELTYP(I).EQ.3) NUMOH(2,IC)=1
3012 CONTINUE
GO TO 3013
3001 CONTINUE
WRITE(6,3015)
3015 FORMAT(/,' FOR THE TOP PLATE INPUT NUMBER OF GROUPS',/,
* ' FOR THE EFFECTIVE FASTENER, LOADED HOLE, UNLOADED',/,
* ' HOLE AND PLAIN ELEMENT',/,
* ' (INPUT 0 IF ELEMENT TYPE IS NOT USED)')
READ(5,M) NEF(1),NLH(1),NOH(1),NPL(1)
WRITE(6,3016)
3016 FORMAT(' GROUPING OF EFFECTIVE FASTENER ELEMENTS:')
N=NEF(1)
DO 3017 I=1,N
WRITE(6,3018) I
3018 FORMAT(' ENTER NUMBER OF ELEMENTS IN GROUP NUMBER',I8)
READ(5,M) NUMEF(1,I)
N1=NUMEF(1,I)
WRITE(6,3019) N1
3019 FORMAT(' ENTER ',I8,' ELEMENT IDS')
READ(5,M) (NOEF(1,I,J),J=1,N1)
3017 CONTINUE
WRITE(6,3020)
3020 FORMAT(/,' GROUPING OF LOADED HOLE ELEMENTS:')
N=NLH(1)
DO 3021 I=1,N
WRITE(6,3021) I
3021 FORMAT(' ENTER NUMBER OF ELEMENTS IN GROUP NUMBER',I8)
READ(5,M) NUMLH(1,I)
N1=NUMLH(1,I)
WRITE(6,3022) N1
3022 FORMAT(' INPUT',I8,' ELEMENT IDS')
READ(5,M) (NGLH(1,I,J),J=1,N1)
3020 CONTINUE
IF(NOH(1).EQ.0) GO TO 4071
WRITE(6,3023)
3023 FORMAT(' GROUPING OF UNLOADED HOLE ELEMENTS')
N=NOH(1)
DO 3024 I=1,N
WRITE(6,3025) I
3025 FORMAT(' ENTER NUMBER OF ELEMENTS IN GROUP NUMBER',I8)
READ(5,M) NUMOH(1,I)

```

```

00005360
00005370
00005380
00005390
00005400
00005410
00005420
00005430
00005440
00005450
00005460
00005470
00005480
00005490
00005500
00005510
00005520
00005530
00005540
00005550
00005560
00005570
00005580
00005590
00005600
00005610
00005620
00005630
00005640
00005650
00005660
00005670
00005680
00005690
00005700
00005710
00005720
00005730
00005740
00005750
00005760
00005770
00005780
00005790
00005800
00005810
00005820
00005830
00005840
00005850
00005860
00005870
00005880
00005890
00005900
00005910
00005920
00005930
00005940
00005950

```

```

      N1=NUMOH(1,I)
      WRITE(6,3026) N1
3026  FORMAT(' ENTER',I8,' ELEMENT IDS')
      READ(5,*) (NGOH(1,I,J),J=1,N1)
3024  CONTINUE
4071  IF(NPL(1).EQ.0) GO TO 4072
      WRITE(6,3027)
3027  FORMAT(' GROUPING OF PLAIN ELEMENTS:')
      N=NPL(1)
      DO 3031 I=1,N
      WRITE(6,3032) I
3032  FORMAT(' ENTER NUMBER OF ELEMENTS IN GROUP NUMBER',I8)
      READ(5,*) NUMPL(1,I)
      N1=NUMPL(1,I)
      WRITE(6,3033) N1
3033  FORMAT(' ENTER',I8,' ELEMENT IDS')
      READ(5,*) (NOPL(1,I,J),J=1,N1)
3031  CONTINUE
4072  CONTINUE
      WRITE(6,4015)
4015  FORMAT(' FOR THE BOTTOM PLATE INPUT NUMBER OF GROUPS',/,
* FOR THE LOADED HOLE, UNLOADED HOLE, AND PLAIN
* ELEMENTS
* (INPUT 0 IF AN ELEMENT TYPE IS NOT USED)')
      READ(5,*) NLH(2),NOH(2),NPL(2)
      NEF(2)=NEF(1)
      N=NEF(1)
      DO 4017 I=1,N
      NUMEF(2,I)=NUMEF(1,I)
      N1=NUMEF(1,I)
      DO 4019 J=1,N1
4019  NOEF(2,I,J)=NOEF(1,I,J)
4017  CONTINUE
      WRITE(6,4088)
4088  FORMAT(' GROUPING OF LOADED HOLE ELEMENTS:')
      N=NLH(2)
      DO 4020 I=1,N
      WRITE(6,4021) I
4021  FORMAT(' ENTER NUMBER OF ELEMENTS IN GROUP NUMBER',I8)
      READ(5,*) NUMLH(2,I)
      N1=NUMLH(2,I)
      WRITE(6,4022) N1
4022  FORMAT(' INPUT',I8,' ELEMENT IDS')
      READ(5,*) (NGLH(2,I,J),J=1,N1)
4020  CONTINUE
      IF(NOH(2).EQ.0) GO TO 4073
      WRITE(6,4023)
4023  FORMAT(' GROUPING OF UNLOADED HOLE ELEMENTS')
      N=NOH(2)
      DO 4024 I=1,N
      WRITE(6,4025) I
4025  FORMAT(' ENTER NUMBER OF ELEMENTS IN GROUP NUMBER',I8)
      READ(5,*) NUMOH(2,I)
      N1=NUMOH(2,I)
      WRITE(6,4026) N1
4026  FORMAT(' ENTER',I8,' ELEMENT IDS')
      READ(5,*) (NGOH(2,I,J),J=1,N1)
4024  CONTINUE
4073  IF(NPL(2).EQ.0) GO TO 4074
      WRITE(6,4027)

```

```

00005960
00005970
00005980
00005990
00006000
00006010
00006020
00006030
00006040
00006050
00006060
00006070
00006080
00006090
00006100
00006110
00006120
00006130
00006140
00006150
00006160
00006170
00006180
00006190
00006200
00006210
00006220
00006230
00006240
00006250
00006260
00006270
00006280
00006290
00006300
00006310
00006320
00006330
00006340
00006350
00006360
00006370
00006380
00006390
00006400
00006410
00006420
00006430
00006440
00006450
00006460
00006470
00006480
00006490
00006500
00006510
00006520
00006530
00006540
00006550

```

```

4027 FORMAT(' GROUPING OF PLAIN ELEMENTS:')
      N=NPL(2)
      DO 4031 I=1,N
      WRITE(6,4032) I
4032 FORMAT(' ENTER NUMBER OF ELEMENTS IN GROUP NUMBER',I5)
      READ(5,*) NUMPL(2,I)
      N1=NUMPL(2,I)
      WRITE(6,4033) N1
4033 FORMAT(' ENTER',I5,' ELEMENT IDS')
      READ(5,*) (NOPL(2,I,J),J=1,N1)
4031 CONTINUE
4074 CONTINUE
3013 CONTINUE
      WRITE(6,3737)
3737 FORMAT(//,' INPUT DATA FOR FAILURE ANALYSIS',//)
      DO 226 K=1,2
      IF(CM(K).NE.CMC) GO TO 2226
      WRITE(6,532) K
532 FORMAT(' ENTER FIBER ULTIMATE STRAIN VALUES ',//,
* ' IN PLATE NO ',I5,//,
* ' EPSILON ULT IN COMPRESSION ',//,
* ' EPSILON ULT IN TENSION ',//,
* ' GAMMA ULT IN SHEAR ',//)
      READ(5,*) (STULT(I,K),I=1,3)
      GO TO 2227
2226 CONTINUE
      WRITE(6,2229)
2229 FORMAT(' ENTER METALLIC STRENGTHS: ',//,
* ' TENSILE STRENGTH ',//,
* ' COMPRESSIVE STRENGTH ',//,
* ' SHEAR STRENGTH')
      READ(5,*) STM(1),STM(2),STM(3)
2227 CONTINUE
      WRITE(6,4054)
4054 FORMAT(//,' AN AVERAGE STRESS CRITERIA IS USED TO ',//,
* ' PREDICT FAILURE. AO VALUES ARE REQUIRED AS ',//,
* ' CHARACTERISTIC DISTANCES OVER WHICH STRESSES ',//,
* ' ARE TO BE AVERAGED AND COMPARED TO UNNOTCHED ',//,
* ' LAMINATE STRENGTHS TO PREDICT FAILURE',//)
      WRITE(6,5432) K
5432 FORMAT(' ENTER AO VALUES FOR STRESS AVERAGING',//,
* ' FOR EACH FAILURE MODE IN PLATE NO',I5,//,
* ' AONT = NET SECTION ',//,
* ' AOBK = BEARING ',//,
* ' AOSO = SHEAROUT ')
      READ(5,*) AONT(K),AOBK(K),AOSO(K)
226 CONTINUE
C
C CASE HEADING
C
      WRITE(6,143)
143 FORMAT(///,10X,'PROGRAM SAMCJ',//)
      IF(NSDLS.EQ.1) WRITE(6,633)
      IF(NCDLS.EQ.2) WRITE(6,634)
633 FORMAT(2X,'A SINGLE LAP SHEAR PANEL WILL BE ANALYZED',//)
634 FORMAT(2X,'A DOUBLE LAP SHEAR PANEL WILL BE ANALYZED',//)
      IF(LTHCM.EQ.1) WRITE(6,823)
      IF(LTHCM.EQ.2) WRITE(6,824)
823 FORMAT(2X,'LOADED IN STATIC TENSION',//)
824 FORMAT(2X,'LOADED IN STATIC COMPRESSION',//)

```

```

00006560
00006570
00006580
00006590
00006600
00006610
00006620
00006630
00006640
00006650
00006660
00006670
00006680
00006690
00006700
00006710
00006720
00006730
00006740
00006750
00006760
00006770
00006780
00006790
00006800
00006810
00006820
00006830
00006840
00006850
00006860
00006870
00006880
00006890
00006900
00006910
00006920
00006930
00006940
00006950
00006960
00006970
00006980
00006990
00007000
00007010
00007020
00007030
00007040
00007050
00007060
00007070
00007080
00007090
00007100
00007110
00007120
00007130
00007140
00007150

```

```

DO 241 I=1,2
WRITE(6,600) I
600 FORMAT(10X,'PLATE NO ',I5,' ',/)
WRITE(6,601) (MTL(I,J),J=1,15)
601 FORMAT(2X,15A4,/)
HT=NELPLS(I,1)*PLYTHK(I,1)
WRITE(6,691) E1(I),E2(I),G12(I),V12(I),V21(I)
691 FORMAT(2X,'MATERIAL PROPERTIES',/,/,
*10X,'E1 = ',D9.3,' PSI',/,
*10X,'E2 = ',D9.3,' PSI',/,
*10X,'G12 = ',D9.3,' PSI',/,
*10X,'NU12 = ',D9.3,/,
*10X,'NU21 = ',D9.3,/)
241 CONTINUE
WRITE(6,606)
606 FORMAT(10X,'FASTENER DESCRIPTION',/)
WRITE(6,607) (MTL(3,J),J=1,15)
607 FORMAT(2X,15A4,/)
WRITE(6,647) FASD
647 FORMAT(2X,' DIAMETER = ',D9.3,' INCHES',/)
WRITE(6,609) FASE,FASV
609 FORMAT(2X,' MATERIAL PROPERTIES',/,/,
*10X,'E = ',D9.3,' PSI',/,
*10X,'MU = ',D9.3,/)
708 CONTINUE
WRITE(6,923)
923 FORMAT(10X,'FAILURE ANALYSIS',/)
WRITE(6,558)
558 FORMAT(2X,' AN AVERAGE STRESS CRITERION WILL BE USED',/)
DO 631 I=1,2
WRITE(6,632) I
632 FORMAT(2X,' PLATE NUMBER',I5,/)
NP=NUMPLY(I)
IF(CM(1,1,1,1)) GO TO 3112
WRITE(6,713)
713 FORMAT(2X,' FIBER STRAIN ULTIMATES',/)
776 WRITE(6,677) (STULT(LL,I),LL=1,3)
677 FORMAT(2X,' EPSILON ULT COMP = ',D9.3,/,
*2X,' EPSILON ULT TEN = ',D9.3,/,
*2X,' GAMMA ULT SHEAR = ',D9.3,/)
GO TO 3113
3112 CONTINUE
WRITE(6,3114)
3114 FORMAT(' METALLIC STRENGTHS ',/)
WRITE(6,3115) STM(1),STM(2),STM(3)
3115 FORMAT(2X,' TENSILE STRENGTH = ',D9.3,/,
*2X,' COMPRESSIVE STRENGTH = ',D9.3,/,
*2X,' SHEAR STRENGTH = ',D9.3,/)
3113 CONTINUE
WRITE(6,1563)
1563 FORMAT(2X,' CHARACTERISTIC DISTANCES',/)
WRITE(6,564) AONT(I),AOBR(I),AOSO(I)
564 FORMAT(' AONT = ',D9.3,' INCHES',/,
* ' AOBR = ',D9.3,' INCHES',/,
* ' AOSO = ',D9.3,' INCHES',/)
631 CONTINUE

```

```

C THE JOINT LOAD DISTRIBUTION IS CALCULATED USING THE
C FINITE ELEMENT METHOD WITH SPECIAL PROBLEM-ADAPTED
C ELEMENTS WHICH EFFECTIVELY REPRESENT THE STIFFNESS

```

```

00007160
00007170
00007180
00007190
00007200
00007210
00007220
00007230
00007240
00007250
00007260
00007270
00007280
00007290
00007300
00007310
00007320
00007330
00007340
00007350
00007360
00007370
00007380
00007390
00007400
00007410
00007420
00007430
00007440
00007450
00007460
00007470
00007480
00007490
00007500
00007510
00007520
00007530
00007540
00007550
00007560
00007570
00007580
00007590
00007600
00007610
00007620
00007630
00007640
00007650
00007660
00007670
00007680
00007690
00007700
00007710
00007720
00007730
00007740
00007750

```


C	PROPERTIES OF FASTENERS, LOADED HOLES, AND OPEN	00007760
C	HOLE REGIONS IN THE JOINT	00007770
C	INTERNAL APPLIED LOAD SET TO 1 KIP	00007780
C	APP=1000.0	00007790
C	IF(LTNCM.EQ.2) APP=-APP	00007800
C	NELTOT=NEL1+NEL2	00007810
C	NGTOT=NGP1+NGP2	00007820
C	INITIALIZE ARRAYS	00007830
C	DO 1 I=1,50	00007840
C	DO 3 J=1,4	00007850
C	DO 3 K=1,4	00007860
C	3 ELSTFF(I,J,K)=0.	00007870
C	1 CONTINUE	00007880
C	DO 4 I=1,200	00007890
C	PBC(I)=0.	00007900
C	HRS(I)=0.	00007910
C	AHR(I)=0.	00007920
C	AHR2(I)=0.	00007930
C	DO 5 J=1,200	00007940
C	GLSTFF(I,J)=0.	00007950
C	ASQM(I,J)=0.	00007960
C	5 CONTINUE	00007970
C	4 CONTINUE	00007980
C	CALCULATION OF EFFECTIVE FASTENER ELEMENT	00007990
C	STIFFNESS MATRICIES	00008000
C	WRITE(6,8418)	00008010
C	8418 FORMAT(/,' PAUSE FOR STIFFNESS MATRIX CALCULATIONS',/)	00008020
C	NLOOP=NEF(1)	00008030
C	DO 444 I=1,NLOOP	00008040
C	NEL=NGEF(1,1)	00008050
C	DO 5001 II=1,NUMF	00008060
C	5001 IF(NEL.EQ.NELFAS(II,1)) IEL=II	00008070
C	SEARCH FOR LOADED HOLE ELEMENTS CONNECTED TO	00008080
C	FASTENER ELEMENT	00008090
C	NTOP=0	00008100
C	NBOT=0	00008110
C	DO 643 J=1,NEL1	00008120
C	643 IF(NELFAS(IEL,2).EQ.NELCON(J,6)) NTOP=J	00008130
C	NP1=NEL1+1	00008140
C	DO 446 J=NP1,NELTOT	00008150
C	446 IF(NELFAS(IEL,3).EQ.NELCON(J,6)) NBOT=J	00008160
C	NPLY(1)=NELPLS(1,LYPN(NTOP))	00008170
C	H(1)=ELTHK(NTOP)	00008180
C	DO 910 JJJ=1,50	00008190
C	910 IPLY(JJJ,1)=NELPT(1,LYPN(NTOP),JJJ)	00008200
C	NPLY(2)=NELPLS(2,LYPN(NBOT))	00008210
C	H(2)=ELTHK(NBOT)	00008220
C	DO 113 JJJ=1,50	00008230
C	113 IPLY(JJJ,2)=NELPT(2,LYPN(NBOT),JJJ)	00008240
C	INITIALIZE PARAMETERS FOR COLLOCATION	00008250
C		00008260
C		00008270
C		00008280
C		00008290
C		00008300
C		00008310
C		00008320
C		00008330
C		00008340
C		00008350

	NT=7	00008360
	NOUT=57	00008370
	NCLL=10	00008380
	NB=NOUT+4*NCLL	00008390
	AX=FSCD(NTOP,3)	00008400
	BX=FSCD(NTOP,3)	00008410
	DO 570 L=1,2	00008420
	PHI=0.000	00008430
	IF(L.EQ.2) PHI=90.00	00008440
	DO 530 K=1,2	00008450
	NTB=NTOP	00008460
	IF(K.EQ.2) NTB=NBOT	00008470
		00008480
	ELEMENT VERTEXES ARE INTERNALLY	00008490
	NUMBERED AS:	00008500
	3 2	00008510
	4 1	00008520
		00003530
		00008540
	SFX=(GCOORD(NELCNA(NTB,5),1)+GCOORD(NELCNA(NTB,2),1))/2.000	00008550
	SFY=(GCOORD(NELCNA(NTB,3),2)+GCOORD(NELCNA(NTB,2),2))/2.000	00008560
	DO 128 JJ=1,4	00008570
	XC(JJ)=GCOORD(NELCNA(NTB,6-JJ),1)-FSCD(NTB,1)	00008580
	YC(JJ)=GCOORD(NELCNA(NTB,6-JJ),2)-FSCD(NTB,2)	00008590
128	CONTINUE	00008600
	XC(5)=XC(1)	00008610
	YC(5)=YC(1)	00008620
	N=ELWDTH(NTB)	00008630
	AST=1000.0	00008640
	CALL POLY(N,AST,JK,K,NCLL,LTNCM)	00008650
	CALL CIRC(N,AST,JK,K,LTNCM)	00008660
	NOPT4=1	00008670
	NCASE=1	00008680
	NTYPE=NELTYP(1EL)	00008690
	CALL FIGEOM(H,PHI,K,NOPT4,NCLL)	00008700
	CALL FBOLT(ANGK,H,PHI,K)	00008710
580	CONTINUE	00008720
	N=NPLY(1)	00008730
	DO 30 II=1,N	00008740
	M=IPLY(II,1)	00008750
30	PLYK(II)=ANGK(M,1)	00008760
	N=NPLY(2)	00008770
	DO 61 II=1,N	00008780
	M1=II+NPLY(1)	00008790
	M2=IPLY(II,2)	00008800
61	PLYK(M1)=ANGK(M2,2)	00008810
		00008820
	CALCULATION OF FASTENER PROPERTIES	00008830
		00008840
	FASG=FASE/(2.*(1+FA5V))	00008850
	FASLAM=5.*(1.0+FA5V)/(7.+6.*FA5V)	00008860
	FASR=FASD/2.	00008870
	FASA=ACOS(-1.)*FASR**2	00008880
	FASI=ACOS(-1.)*FASR**4/4.	00008890
	FASSS=FASLAM*FASG*FASA	00008900
	FASBS=FASE*FASI	00008910
	P=1000.	00008920
	CALL CENTD(H,FASSS,FASBS,P)	00008930
	CALL SOLVE(H,P,U1,U2)	00008940
	IF(L.EQ.2) GO TO 666	00008950

	RDSTFF(IEL,1)=DABS(P/(U1+U2))	00008960
	GO TO 570	00008970
666	RDSTFF(IEL,2)=DABS(P/(U1+U2))	00008980
570	CONTINUE	00008990
	IF(NUMEF(1,I).EQ.1) GO TO 444	00009000
	N=NUMEF(1,1)	00009010
	DO 5023 K=2,N	00009020
	DO 5024 L=1,NUM?	00009030
5024	IF(NGEF(1,I,K).EQ.NELFAS(L,1)) IEL2=L	00009040
	RDSTFF(IEL2,1)=RDSTFF(IEL,1)	00009050
	RDSTFF(IEL2,2)=RDSTFF(IEL,2)	00009060
5023	CONTINUE	00009070
444	CONTINUE	00009080
8584	CONTINUE	00009090
C		00009100
C	CALCULATION OF LOADED HOLE AND UNLOADED HOLE	00009110
C	ELEMENT STIFFNESS MATRICIES	00009120
C		00009130
C		00009140
C	INITIALIZE GAUSSIAN QUADRATURE POINTS AND WEIGHTS	00009150
	NGP=5	00009160
	GSSX(1)=-0.9739065285	00009170
	GSSX(2)=-0.8650633666	00009180
	GSSX(3)=-0.6794095682	00009190
	GSSX(4)=-0.4333953941	00009200
	GSSX(5)=-0.1488743389	00009210
	GSSW(1)=0.0666713443	00009220
	GSSW(2)=0.1494513491	00009230
	GSSW(3)=0.2190863625	00009240
	GSSW(4)=0.2492467193	00009250
	GSSW(5)=0.295542247	00009260
	DO 588 I=1,NGP	00009270
	GSSX(I+NGP)=-GSSX(NGP-I+1)	00009280
	GSSW(I+NGP)=GSSW(NGP-I+1)	00009290
588	CONTINUE	00009300
	NAVD=10	00009310
	DO 420 KJ=1,2	00009320
	ISLH=1	00009330
	ISOH=1	00009340
	ISPL=1	00009350
	NLOOP=NLH(KJ)+NOH(KJ)+NPL(KJ)	00009360
	NCLH=0	00009370
	NCOH=0	00009380
	NCPL=0	00009390
	DO 400 L=1,NLOOP	00009400
	IF(NCLH.EQ.NLH(KJ)) ISLH=0	00009410
	IF(NCLH.EQ.NLH(KJ)) GO TO 6010	00009420
	NCLH=NCLH+1	00009430
	IEL2=NOLH(KJ,NCLH,1)	00009440
	GO TO 6011	00009450
6010	IF(NCOH.EQ.NOH(KJ)) ISOH=0	00009460
	IF(NCOH.EQ.NOH(KJ)) GO TO 6020	00009470
	NCOH=NCOH+1	00009480
	IEL2=NOOH(KJ,NCOH,1)	00009490
	GO TO 6011	00009500
6020	IF(NCPL.EQ.NPL(KJ)) ISPL=0	00009510
	IF(NCPL.EQ.NPL(KJ)) GO TO 400	00009520
	NCPL=NCPL+1	00009530
	IEL2=NOPI(KJ,NCPL,1)	00009540
		00009550

6011	CONTINUE	00009560
	DO 6030 KK=1,NELTOT	00009570
6030	IF(IEL2.EQ.NELCON(KK,1)) IEL=KK	00009580
	H(KJ)=ELTHK(IEL)	00009590
	NPLY(KJ)=NELPLS(KJ,LYPN(IEL))	00009600
	DO 919 JJJ=1,50	00009610
	IPLY(JJJ,KJ)=NELPT(KJ,LYPN(IEL),JJJ)	00009620
919	CONTINUE	00009630
	NRNK=5	00009640
	IF(NELTYP(IEL).EQ.2) NRNK=7	00009650
	INTERNAL NUMBERING OF ELEMENT VERTICIES:	00009660
	3 2	00009670
	4 1	00009680
	SFX=(GCOORD(NELCHA(IEL,5),1)+GCOORD(NELCHA(IEL,2),1))/2.000	00009690
	SFY=(GCOORD(NELCHA(IEL,3),2)+GCOORD(NELCHA(IEL,2),2))/2.000	00009700
	DO 440 K=1,4	00009710
	XC(K)=GCOORD(NELCHA(IEL,6-K),1)-FSCD(IEL,1)	00009720
	IF(NELTYP(IEL).EQ.1) XC(K)=GCOORD(NELCHA(IEL,6-K),1)-SFX	00009730
	YC(K)=GCOORD(NELCHA(IEL,6-K),2)-FSCD(IEL,2)	00009740
	IF(NELTYP(IEL).EQ.1) YC(K)=GCOORD(NELCHA(IEL,6-K),2)-SFY	00009750
440	CONTINUE	00009760
	XC(5)=XC(1)	00009770
	YC(5)=YC(1)	00009780
	AX=FSCD(IEL,3)	00009790
	IF(NELTYP(IEL).EQ.1) AX=0.1	00009800
	BX=AX	00009810
	PI=DARCO(-1.000)	00009820
	RAD=PI/180.00	00009830
	NOAUSS=2*NGP	00009840
	NGPT=4*NOAUSS*2	00009850
	IC=0	00009860
	NCPT=2*NGP	00009870
	DETERMINE COORDINATES AT WHICH STRESSES AND	00009880
	DISPLACEMENTS ARE TO BE COMPUTED.	00009890
	ELEMENT NATURAL FLEXIBILITY MATRICES	00009900
	ARE COMPUTED BY INTEGRATING STRESSES	00009910
	FOR EACH LOAD CASE IN THE NATURAL	00009920
	MODE METHOD. THE ELEMENTS ARE DIVIDED	00009930
	INTO FOUR REGIONS AND THE GAUSSIAN POINTS	00009940
	ARE SCALED TO EACH REGION SIZE	00009950
	REGION 1	00009960
	DO 15 II=1,NCPT	00009970
	DO 15 JJ=1,NCPT	00009980
	IC=IC+1	00009990
	XOUT(IC)=((-AX-XC(1))/2.)*OSSX(II)+(-AX+XC(3))/2.	00010000
	YOUT(IC)=((YC(3)-YC(4))/2.)*OSSX(JJ)+(YC(3)+YC(4))/2.	00010010
	WOHT(IC)=OSSW(II)*OSSW(JJ)*(YC(3)-YC(4))*(-AX-XC(3))/4.00	00010020
15	CONTINUE	00010030
	REGION 2	00010040
		00010050
		00010060
		00010070
		00010080
		00010090
		00010100
		00010110
		00010120
		00010130
		00010140
		00010150

	DO 16 II=1,NCPT	00010160
	DO 16 JJ=1,NCPT	00010170
	IC=IC+1	00010180
	XOUT(IC)=AX*GSSX(II)	00010190
	YI=DSQRT(AX**2-XOUT(IC)**2)	00010200
	YOUT(IC)=((YC(2)-YI)/2.)*GSSX(JJ)+(YC(2)+YI)/2.	00010210
	WOHT(IC)=GSSW(II)*GSSW(JJ)*(YC(2)-YI)*AX/2.0D0	00010220
16	CONTINUE	00010230
C		00010240
C	REGION 3	00010250
C		00010260
	DO 17 II=1,NCPT	00010270
	DO 17 JJ=1,NCPT	00010280
	IC=IC+1	00010290
	XOUT(IC)=AX*GSSX(II)	00010300
	YI=-DSQRT(AX**2-XOUT(IC)**2)	00010310
	YOUT(IC)=((YI-YC(1))/2.)*GSSX(JJ)+(YI+YC(1))/2.	00010320
	WOHT(IC)=GSSW(II)*GSSW(JJ)*(YI-YC(1))*AX/2.0D0	00010330
17	CONTINUE	00010340
C		00010350
C	REGION 4	00010360
C		00010370
	DO 18 II=1,NCPT	00010380
	DO 18 JJ=1,NCPT	00010390
	IC=IC+1	00010400
	XOUT(IC)=((XC(1)-AX)/2.)*GSSX(II)+(XC(1)+AX)/2.	00010410
	YOUT(IC)=((YC(2)-YC(1))/2.)*GSSX(JJ)+(YC(2)+YC(1))/2.	00010420
	WOHT(IC)=GSSW(II)*GSSW(JJ)*(YC(2)-YC(1))*(XC(1)-AX)/4.0D0	00010430
18	CONTINUE	00010440
	NINT=IC	00010450
	N=4*(NCPT**2)	00010460
C		00010470
C	ADD COORDINATES ALONG WHICH STRESSES WILL	00010480
C	BE AVERAGED	00010490
C		00010500
	ANT=AONT(KJ)	00010510
	ABR=AOBR(KJ)	00010520
	ASO=AOSI(KJ)	00010530
	SG=1.0	00010540
	IF(LTHCM.EQ.2) SG=-1.0	00010550
	IF(KJ.EQ.2) SG=-SG	00010560
C		00010570
C	NET SECTION	00010580
C		00010590
	ANDO=ANT/FLOAT(NAVD)	00010600
	DO 21 II=1,NAVD	00010610
	IC=IC+1	00010620
	XOUT(IC)=0.0D0	00010630
	YOUT(IC)=BX+ANDO/2.+(II-1)*ANDO	00010640
21	CONTINUE	00010650
C		00010660
C	SHEAROUT	00010670
C		00010680
	ANSO=ASO/FLOAT(NAVD)	00010690
	DO 31 II=1,NAVD	00010700
	IC=IC+1	00010710
	XOUT(IC)=SG*(BX+ANSO/2.+(II-1)*ANSO)	00010720
	YOUT(IC)=BX	00010730
31	CONTINUE	00010740
C		00010750

C	BEARING	00010760
C	ANBR=ABR/FLOAT(NAVD)	00010770
	DO 41 II=1,NAVD	00010780
	IC=IC+1	00010790
	XOUT(IC)=30*(AX+ANBR/2.+(II-1)*ANBR)	00010800
	YOUT(IC)=0.	00010810
41	CONTINUE	00010820
C	ADD COORDINATES ALONG WHICH ELEMENT LOAD	00010830
C	RECOVERY WILL BE COMPUTED	00010840
C	DO 3332 III=1,10	00010850
	IC=IC+1	00010860
	IF(KJ.EQ.1) XOUT(IC)=XC(3)+0.1*AX	00010870
	IF(KJ.EQ.2) XOUT(IC)=XC(1)-0.1*AX	00010880
	YOUT(IC)=((YC(2)-YC(1))/2.000)*OSSX(III)+(YC(2)+YC(1))/2.000	00010890
C	STRESSES ARE SINGULAR AT THETA = 180 DEG OR Y = 0	00010900
C	IF(DABS(YOUT(IC)).LT.0.01) YOUT(IC)=YOUT(IC-1)	00010910
3332	CONTINUE	00010920
4891	CONTINUE	00010930
	NSTS=4*NAVD	00010940
	NOUT=4*(NOAUSS**2)	00010950
C	CALCULATION OF LOADED HOLE, UNLOADED HOLE, AND	00010960
C	PLAIN ELEMENT STIFFNESS MATRICIES	00010970
C	THN=DARCOS(-1.000)/FLOAT(NOP)	00010980
	NN=IEL	00010990
	DO 410 J=1,NNRK	00011000
	NOPT4=5	00011010
	NT=7	00011020
	NCLL=10	00011030
	NB=52+4*NCLL	00011040
	HT=H(KJ)*NPLY(KJ)	00011050
	HCASE=J	00011060
	NTYPE=NELTYP(IEL)	00011070
	CALL MOED(HT,W,AST,J,NN,KJ,NEL,NCLL)	00011080
	CALL MCIR(W,AST,NN,J,NCLL)	00011090
	PHI=0.000	00011100
	CALL AMATRX(H,PHI,KJ)	00011110
	CALL FIGEOM(H,PHI,KJ,NOPT4,NCLL)	00011120
	CALL INFLN(WOHT,H,NNRK,J,KJ,NN,NOPT)	00011130
410	CONTINUE	00011140
4000	CONTINUE	00011150
C	COMPUTE ELEMENT FAILURE VALUES BASED	00011160
C	ON MAXIMUM FIBER STRAIN ALLOWABLES	00011170
C	HT=H(KJ)*NPLY(KJ)	00011180
	IF(NELTYP(IEL).EQ.2) CALL SMAX(HT,KJ,IEL)	00011190
	IF(NELTYP(IEL).EQ.3) CALL SMAX(HT,KJ,IEL)	00011200
	IF(ISLH.EQ.0) GO TO 6040	00011210
	NL=NUMLH(KJ,NCLH)	00011220
	IF(NL.EQ.1) GO TO 400	00011230
	DO 6041 K=2,NL	00011240
	DO 6042 LL=1,NELTOT	00011250
6042	IF(NOLH(KJ,NCLH,K).EQ.NELCON(LL,1)) IEL2=LL	00011260
		00011270
		00011280
		00011290
		00011300
		00011310
		00011320
		00011330
		00011340
		00011350

DO 6043 ILM=1,10	00011360
DO 6043 ILK=1,10	00011370
6043 ELSTFF(IEL2,ILM,ILK)=ELSTFF(IEL,ILM,ILK)	00011380
DO 6044 KK=1,4	00011390
6044 PSMX(IEL2,KK)=PSMX(IEL,KK)	00011400
NNN=4*NAVD	00011410
DO 6045 ILM=1,NNN	00011420
DO 6045 ILK=1,10	00011430
6045 ELSTSS(IEL2,ILM,ILK)=ELSTSS(IEL,ILM,ILK)	00011440
6041 CONTINUE	00011450
GO TO 400	00011460
6040 IF(ISOH.EQ.0) GO TO 6046	00011470
NL=NUMOH(KJ,NGOH)	00011480
IF(NL.EQ.1) GO TO 400	00011490
DO 6047 K=2,NL	00011500
DO 6048 LL=1,NELTOT	00011510
6048 IF(NGOH(KJ,NGOH,K).EQ.NELCON(LL,1)) IEL2=LL	00011520
DO 6049 ILM=1,10	00011530
DO 6049 ILK=1,10	00011540
6049 ELSTFF(IEL2,ILM,ILK)=ELSTFF(IEL,ILM,ILK)	00011550
DO 6050 KK=1,4	00011560
6050 PSMX(IEL2,KK)=PSMX(IEL,KK)	00011570
NNN=4*NAVD	00011580
DO 6051 ILM=1,NNN	00011590
DO 6051 ILK=1,10	00011600
6051 ELSTSS(IEL2,ILM,ILK)=ELSTSS(IEL,ILM,ILK)	00011610
6047 CONTINUE	00011620
GO TO 400	00011630
6046 IF(ISPL.EQ.0) GO TO 400	00011640
NL=NUMPL(KJ,NCPL)	00011650
IF(NL.EQ.1) GO TO 400	00011660
DO 6053 K=2,NL	00011670
DO 6054 LL=1,NELTOT	00011680
6054 IF(NGPL(KJ,NCPL,K).EQ.NELCON(LL,1)) IEL2=LL	00011690
DO 6055 ILM=1,10	00011700
DO 6055 ILK=1,10	00011710
6055 ELSTFF(IEL2,ILM,ILK)=ELSTFF(IEL,ILM,ILK)	00011720
DO 6056 KK=1,4	00011730
6056 PSMX(IEL2,KK)=PSMX(IEL,KK)	00011740
NNN=4*NAVD	00011750
DO 6057 ILM=1,NNN	00011760
DO 6057 ILK=1,10	00011770
6057 ELSTSS(IEL2,ILM,ILK)=ELSTSS(IEL,ILM,ILK)	00011780
6053 CONTINUE	00011790
400 CONTINUE	00011800
420 CONTINUE	00011810
C DETERMINE ELEMENT ARRANGEMENT IN TOP	00011820
C AND BOTTOM PLATES	00011830
C	00011840
DO 681 KJ=1,2	00011850
IF(KJ.EQ.2) GO TO 501	00011860
L1=1	00011870
L2=NGP1	00011880
L3=1	00011890
L4=NEL1	00011900
GO TO 502	00011910
501 L1=NGP1+1	00011920
L2=NGTOT	00011930
L3=NEL1+1	00011940
	00011950

```

      L4=NELTOT
502  CONTINUE
      AXMIN=1.D10
      AYMIN=1.D10
      DO 503 I=1, L2
        IF(AXMIN.GT.GCOORD(I,1)) AXMIN=GCOORD(I,1)
        IF(AYMIN.GT.GCOORD(I,2)) AYMIN=GCOORD(I,2)
        IF(AXMIN.EQ.GCOORD(I,1).AND.AYMIN.EQ.GCOORD(I,2)) NC=I
503  CONTINUE
      DO 574 I=1, L4
        IF(NELCON(I,2).EQ.NORID(NC)) IEL=I
574  IF(NELCON(I,2).EQ.NORID(NC)) IEL=I
        NELGRD(KJ,1,1)=IEL
        DO 504 I=1, 25
          DO 505 J=1, 25
            IFL=0
            DO 506 K=1, L4
              IF(NELCON(K,2).EQ.NELCON(NELORD(KJ,J,1),3)) IEL=K
              IF(IEI.EQ.0) GO TO 507
              NELORD(KJ,J+1,1)=IEL
505  CONTINUE
507  CONTINUE
              IF(KJ.EQ.1) NROW1=J
              IF(KJ.EQ.2) NROW2=J
              IEL=0
              DO 508 L=1, L4
                IF(NELCON(NELORD(KJ,1,1),5).EQ.NELCON(
                    L,2)) IEL=L
                IF(IEI.EQ.0) GO TO 507
                NELORD(KJ,1,1+1)=IEL
504  CONTINUE
509  CONTINUE
                IF(KJ.EQ.1) NCOL1=I
                IF(KJ.EQ.2) NCOL2=I
681  CONTINUE
      CCG
      COMPUTE NODAL DEGREES OF FREEDOM
      IC=0
      DO 540 KJ=1, 2
        IF(KJ.EQ.1) NR=NROW1
        IF(KJ.EQ.1) NC=NCOL1
        IF(KJ.EQ.2) NR=NROW2
        IF(KJ.EQ.2) NC=NCOL2
        NELD1S(NELORD(KJ,1,1),1,1)=IC+1
        NELD1S(NELORD(KJ,1,1),1,2)=IC+2
        NELD1S(NELORD(KJ,1,1),2,1)=IC+3
        NELD1S(NELORD(KJ,1,1),2,2)=IC+4
        IC=IC+4
        IF(NR.EQ.1) GO TO 549
        DO 541 I=2, NR
          NELD1S(NELORD(KJ,I,1),1,1)=NELDIS(NELORD(KJ,I-1
              ,1),2,1)
          NELD1S(NELORD(KJ,I,1),1,2)=NELDIS(NELORD(KJ,I-1
              ,1),2,2)
          NELD1S(NELORD(KJ,I,1),2,1)=IC+1
          NELD1S(NELORD(KJ,I,1),2,2)=IC+2
          IC=IC+2
541  CONTINUE
549  CONTINUE
        DO 542 I=1, NC

```

```

00011960
00011970
00011980
00011990
00012000
00012010
00012020
00012030
00012040
00012050
00012060
00012070
00012080
00012090
00012100
00012110
00012120
00012130
00012140
00012150
00012160
00012170
00012180
00012190
00012200
00012210
00012220
00012230
00012240
00012250
00012260
00012270
00012280
00012290
00012300
00012310
00012320
00012330
00012340
00012350
00012360
00012370
00012380
00012390
00012400
00012410
00012420
00012430
00012440
00012450
00012460
00012470
00012480
00012490
00012500
00012510
00012520
00012530
00012540
00012550

```


DO 543 J=1,NR	00012560
IF(I.EQ.1) GO TO 544	00012570
1 NELDIS(NELORD(KJ,J,I),1,1)=NELDIS(NELORD(KJ,J,I-1),4,1)	00012580
NELDIS(NELORD(KJ,J,I),1,2)=NELDIS(NELORD(KJ,J,I-1),4,2)	00012590
NELDIS(NELORD(KJ,J,I),2,1)=NELDIS(NELORD(KJ,J,I-1),3,1)	00012600
NELDIS(NELORD(KJ,J,I),2,2)=NELDIS(NELORD(KJ,J,I-1),3,2)	00012610
544 CONTINUE	00012620
IF(J.EQ.1) GO TO 561	00012630
NELDIS(NELORD(KJ,J,I),4,1)=NELDIS(NELORD(KJ,J-1,I),3,1)	00012640
NELDIS(NELORD(KJ,J,I),4,2)=NELDIS(NELORD(KJ,J-1,I),3,2)	00012650
GO TO 562	00012660
561 CONTINUE	00012670
NELDIS(NELORD(KJ,J,I),4,1)=IC+1	00012680
NELDIS(NELORD(KJ,J,I),4,2)=IC+2	00012690
IC=IC+2	00012700
562 CONTINUE	00012710
IF(HELTY(NELORD(KJ,J,I)).NE.2) GO TO 543	00012720
NELDIS(NELORD(KJ,J,I),5,1)=IC+1	00012730
NELDIS(NELORD(KJ,J,I),5,2)=IC+2	00012740
IC=IC+2	00012750
543 CONTINUE	00012760
NELDIS(NELORD(KJ,J,I),3,1)=IC+1	00012770
NELDIS(NELORD(KJ,J,I),3,2)=IC+2	00012780
IC=IC+2	00012790
543 CONTINUE	00012800
542 CONTINUE	00012810
540 CONTINUE	00012820
C DETERMINE BOUNDARY NODES AND VALUES	00012830
C	00012840
C	00012850
NRD=2*(NGP1+NGP2)	00012860
DO 165 I=1,100	00012870
165 PBC(I)=0.	00012880
C DISTRIBUTE APPLIED LOAD	00012890
C	00012900
C	00012910
ATOT=GCOORD(NELCNA(NELORD(1,NROW1,1),3),2)	00012920
X-GCOORD(NELCNA(NELORD(1,1,1),2),2)	00012930
APL=APP/ATOT	00012940
SG=1.0	00012950
IF(LTHCM.EQ.1) SG=-1.0	00012960
DO 178 I=1,NROW1	00012970
A1=GCOORD(NELCNA(NELORD(1,I,1),3),2)	00012980
X-GCOORD(NELCNA(NELORD(1,I,1),2),2)	00012990
M1=NELDIS(NELORD(1,I,1),1,1)	00013000
M2=NELDIS(NELORD(1,I,1),2,1)	00013010
PBC(M1)=PBC(M1)+SG*(0.5*DABS(APL*M1))	00013020
PBC(M2)=PBC(M2)+SG*(0.5*DABS(APL*M2))	00013030
178 CONTINUE	00013040
1119 CONTINUE	00013050
C ASSEMBLE GLOBAL STIFFNESS MATRIX	00013060
C	00013070
C	00013080
DO 220 N1=1,NELTOT	00013090
IR=5	00013100
IF(HELTY(N1).NE.2) IR=4	00013110
	00013120
	00013130
	00013140
	00013150

C	TOP AND BOTTOM PLATE LOADED HOLE AND	00013160
C	UNLOADED HOLE ELEMENTS	00013170
C		00013180
	IC1=0	00013190
	DO 425 N2=1,IR	00013200
	DO 425 N3=1,2	00013210
	M1=NELDIS(N1,N2,N3)	00013220
	IC1=IC1+1	00013230
	IC2=0	00013240
	DO 425 N4=1,IR	00013250
	DO 425 N5=1,2	00013260
	M2=NELDIS(N1,N4,N5)	00013270
	IC2=IC2+1	00013280
	OLSTFF(M1,M2)=OLSTFF(M1,M2)+ELSTFF(N1,IC1,IC2)	00013290
425	CONTINUE	00013300
220	CONTINUE	00013310
C		00013320
C	ADD EFFECTIVE FASTENER ELEMENTS	00013330
	DO 260 I=1,NUMF	00013340
	DO 1541 J=1,NEL1	00013350
1541	IF(NELFAS(I,2).EQ.NELCON(J,6)) N=J	00013360
	N1=NELDIS(N,5,1)	00013370
	N2=NELDIS(N,5,2)	00013380
	NL=NEL1+1	00013390
	DO 1562 J=NL,NELTOT	00013400
1562	IF(NELFAS(I,3).EQ.NELCON(J,6)) N=J	00013410
	N3=NELDIS(N,5,1)	00013420
	N4=NELDIS(N,5,2)	00013430
	OLSTFF(N1,N1)=OLSTFF(N1,N1)+RDSTFF(I,1)	00013440
	OLSTFF(N1,N3)=OLSTFF(N1,N3)+RDSTFF(I,1)	00013450
	OLSTFF(N2,N2)=OLSTFF(N2,N2)+RDSTFF(I,2)	00013460
	OLSTFF(N2,N4)=OLSTFF(N2,N4)+RDSTFF(I,2)	00013470
	OLSTFF(N3,N3)=OLSTFF(N3,N3)+RDSTFF(I,1)	00013480
	OLSTFF(N3,N1)=OLSTFF(N3,N1)+RDSTFF(I,1)	00013490
	OLSTFF(N4,N4)=OLSTFF(N4,N4)+RDSTFF(I,2)	00013500
	OLSTFF(N4,N2)=OLSTFF(N4,N2)+RDSTFF(I,2)	00013510
260	CONTINUE	00013520
	NP=2*(NQP1+NQP2)	00013530
C		00013540
C	GLOBAL BOUNDARY CONDITIONS	00013550
C		00013560
	DO 415 I=1,NP	00013570
	RHS(I)=PBC(I)	00013580
415	CONTINUE	00013590
C		00013600
	IC=1	00013610
	NZERO(IC)=NELDIS(NELORD(2,1,NCOL2),4,1)	00013620
	DO 437 I=1,NROW2	00013630
	IC=IC+1	00013640
	NZERO(IC)=NELDIS(NELORD(2,1,NCOL2),3,1)	00013650
437	CONTINUE	00013660
	IC=IC+1	00013670
	NZERO(IC)=NELDIS(NELORD(2,1,NCOL2),4,2)	00013680
	NUMZ=IC	00013690
C		00013700
C	RESTORE REDUCED STIFFNESS MATRIX	00013710
C		00013720
	ICF=0	00013730
		00013740
		00013750

	GO TO 2011	00014360
3112	CONTINUE	00014370
	ANR(1)=0.	00014380
2011	CONTINUE	00014390
C		00014400
C	CALCULATE NODAL LOADS	00014410
C		00014420
	IC=0	00014430
	DO 44 I=1,NRD	00014440
	DO 54 J=1,NUMZ	00014450
	IF(I.NE.NZERO(J)) GO TO 54	00014460
	ANR2(I)=0.0D0	00014470
	GO TO 44	00014480
54	CONTINUE	00014490
	IC=IC+1	00014500
	ANR2(I)=ANR(IC)	00014510
44	CONTINUE	00014520
	WRITE(6,3712)	00014530
3712	FORMAT(/,10X,'ELEMENT FORCES',//)	00014540
	DO 500 K=1,HELTOF	00014550
	NID=HELCON(K,1)	00014560
	WRITE(6,3947) NID	00014570
8947	FORMAT(/,' ELEMENT ID',IS,/,	00014580
	*6X,'GRID',9X,'FX',9X,'FY',/)	00014590
	IR=5	00014600
	KL=K	00014610
	IF(KL.GT.NE1) AL=K-NEL	00014620
	IF(NE1YP(K).NE.2) IR=4	00014630
	DO 510 I=1,IR	00014640
	SUMU=0.	00014650
	SUMV=0.	00014660
	N=2*I-1	00014670
	DO 520 J=1,IR	00014680
	N1=HELDIS(K,J,1)	00014690
	N2=HELDIS(K,J,2)	00014700
	SUMU=SUMU+ELSTFF(K,N,(2*J-1))*ANR2(N1)+	00014710
	*ELSTFF(K,N,(2*J))*ANR2(N2)	00014720
520	CONTINUE	00014730
	N=2*I	00014740
	DO 530 J=1,IR	00014750
	N1=HELDIS(K,J,1)	00014760
	N2=HELDIS(K,J,2)	00014770
	SUMV=SUMV+ELSTFF(K,N,(2*J-1))*ANR2(N1)+	00014780
	*ELSTFF(K,N,(2*J))*ANR2(N2)	00014790
530	CONTINUE	00014800
C		00014810
C	STORE ELEMENT LOADS FOR CHECK ON ELEMENT	00014820
C	LOAD RECOVERY	00014830
C		00014840
	IF(K.LE.NEL1.AND.(I.EQ.1.OR.I.EQ.2)) ELLOAD(K,I)=SUMU	00014850
	IF(K.GT.NEL1.AND.(I.EQ.3.OR.I.EQ.4)) ELLOAD(K,I-2)=SUMU	00014860
	NID=HELCON(K,I+1)	00014870
	WRITE(6,3239) NID,SUMU,SUMV	00014880
3239	FORMAT(2X,I8,5X,2(D9.3,2X))	00014890
510	CONTINUE	00014900
500	CONTINUE	00014910
C		00014920
C	COMPUTE ELEMENT FAILURE LOADS AND DETERMINE	00014930
C	CRITICAL ELEMENT TO CALCULATE JOINT FAILURE	00014940
C	LOAD	00014950

C	CALL FCRT(APP,NEL1,NEL2,NDAM,IN,LTNCM,NAVD)	00014960
	FAILV=DABS(ELFAIL(IN,NDAM))	00014970
	IF(NSDLS.EQ.2) FAILV=2.*FAILV	00014980
	NID=NELCON(IN,1)	00014990
	WRITE(6,5555) NID,FAILV	00015000
5555	FORMAT(///,' FAILURE IS PREDICTED TO OCCUR IN ELEMENT',/,	00015010
	' NUMBER',15,' AT AN APPLIED JOINT LOAD VALUE ',/,	00015020
	' OF ',D14.7,' LBS',/)	00015030
	IF(NDAM.EQ.1) WRITE(6,5556)	00015040
	IF(NDAM.EQ.2) WRITE(6,5557)	00015050
	IF(NDAM.EQ.3) WRITE(6,5558)	00015060
5556	FORMAT(' THE PREDICTED FAILURE MODE IS NET SECTION')	00015070
5557	FORMAT(' THE PREDICTED FAILURE MODE IS SHEAR-OUT')	00015080
5558	FORMAT(' THE PREDICTED FAILURE MODE IS BEARING')	00015090
	STOP	00015100
	END	00015110
C		00015120
C		00015130
C		00015140
C	SUBROUTINE MGEO(HT,W,AST,J,IN,KJ,NEL,NCL)	00015150
	IMPLICIT REAL*8(A-H,O-Z)	00015160
	DIMENSION A1A(4),A2A(4)	00015170
	DIMENSION XB(200),YB(200),A1(200),A2(200)	00015180
	DIMENSION NELTYP(50),THTA(200)	00015190
	DIMENSION XC(5),YC(5)	00015200
	COMMON/CMT1/XB,YB,A1,A2,THTA	00015210
	COMMON/XCYC/XC,YC	00015220
	COMMON/NTP/NELTYP	00015230
C		00015240
C		00015250
C	DETERMINE EXTERIOR COLLOCATION POINTS AND	00015260
C	STRESS BOUNDARY CONDITIONS CORRESPONDING	00015270
C	TO THE NATURAL LOAD CASES	00015280
	NCS=5	00015290
	IF(NELTYP(IN).NE.2) NCS=3	00015300
	JK=0	00015310
	DO 15 I=1,4	00015320
	A1A(I)=0.	00015330
15	A2A(I)=0.	00015340
	A=(YC(2)-YC(1))*HT	00015350
	B=(XC(2)-XC(1))*HT	00015360
	IF(J.EQ.1) A1A(3)=1.0D0/A	00015370
	IF(J.EQ.1.AND.NELTYP(IN).NE.2) A1A(1)=1.0D0/A	00015380
	IF(J.EQ.2) A1A(2)=1.0D0/B	00015390
	IF(J.EQ.2.AND.NELTYP(IN).NE.2) A1A(4)=1.0D0/B	00015400
	IF(J.EQ.2.AND.NELTYP(IN).NE.2) GO TO 55	00015410
	IF(J.EQ.3) A1A(1)=1.0D0/A	00015420
	IF(J.EQ.4) A1A(4)=1.0D0/B	00015430
	IF(J.EQ.1.OR.J.EQ.3) AST=1.0D0/A	00015440
	IF(J.EQ.2.OR.J.EQ.4) AST=1.0D0/B	00015450
55	CONTINUE	00015460
	H=XC(3)-XC(2)	00015470
	IF(J.EQ.1.OR.J.EQ.3.OR.J.EQ.6) H=YC(2)-YC(1)	00015480
	DO 10 I=1,4	00015490
	X=XC(I)-XC(I+1)	00015500
	Y=YC(I+1)-YC(I)	00015510
	IF(X.EQ.0.) X=1.D-6	00015520
	IF(Y.EQ.0.) Y=1.D-6	00015530
		00015540
		00015550

	TH=DATAN2(X,Y)	00015560
	TH=TH*180./DARCOS(-0.1D1)	00015570
	DX=(XC(I+1)-XC(I))/(NCL+1)	00015580
	DY=(YC(I+1)-YC(I))/(NCL+1)	00015590
	DO 20 II=1,NCL	00015600
	JK=JK+1	00015610
	IF(I.EQ.1.OR.I.EQ.3) GO TO 23	00015620
	YB(JK)=YC(I)	00015630
	XB(JK)=XC(I)+DX*(II+5)	00015640
	IF(II.EQ.1) XB(JK)=XC(I)+(DX/2.)	00015650
	GO TO 24	00015660
23	YB(JK)=YC(I)+DY*(II+5)	00015670
	IF(II.EQ.1) YB(JK)=YC(I)+(DY/2.)	00015680
	XB(JK)=XC(I)	00015690
24	THTA(JK)=TH	00015700
	A1(JK)=A1A(I)	00015710
	A2(JK)=A2A(I)	00015720
	IF(J.EQ.NCS.AND.(I.EQ.1.OR.I.EQ.3)) A1(JK)=(2*YB(JK)/W)*(3.0D0/A)	00015730
	IF(J.EQ.(NCS+1).AND.(I.EQ.2.OR.I.EQ.4)) A1(JK)=(2*XB(JK)/W)*(3.0	00015740
	*D0/3)	00015750
	IF(J.EQ.(NCS+2).AND.(I.EQ.1.OR.I.EQ.3)) A2(JK)=2.0/DSQRT(A**2+B**2	00015760
	*X)	00015770
	IF(J.EQ.(NCS+2).AND.(I.EQ.2.OR.I.EQ.4)) A2(JK)=-2.0/DSQRT(A**2+B**2	00015780
	*X2)	00015790
20	CONTINUE	00015800
10	CONTINUE	00015810
	RETURN	00015820
	END	00015830
C		00015840
C		00015850
C		00015860
	SUBROUTINE MCIR(W,AST,I,J,NCL)	00015870
	IMPLICIT REAL*8(A-H,O-Z)	00015880
	DIMENSION XB(200),YB(200),A1(200),A2(200)	00015890
	DIMENSION THTA(200),NELTYP(50)	00015900
	COMMON/CNT1/XB,YB,A1,A2,THTA	00015910
	COMMON/NTP/NELTYP	00015920
	COMMON/ELP/AX,DX	00015930
	CCN=-1.0D0	00015940
	RAD=DARCOS(CON)/180.	00015950
	BSTR=DABS((2.*W*AST)/(DARCOS(CON)*AX))	00015960
	IF(NELTYP(I).NE.2) BSTR=0.0D0	00015970
	IF(NELTYP(I).EQ.2.AND.J.GT.5) BSTR=0.0D0	00015980
	IC=4*NCL	00015990
		00016000
C	DETERMINE INTERIOR COLLOCATION POINTS AND	00016010
C	STRESS BOUNDARY CONDITIONS	00016020
C		00016030
	NBD=52	00016040
	NBI=NBD/4	00016050
	DO 1) K=1,4	00016060
	CR=(K-1)*DARCOS(CON)/2.	00016070
	DO 20 KI=1,NBI	00016080
	IC=IC+1	00016090
	THINC=(DARCOS(CON)/2.)/FLOAT(NBI)	00016100
	A1(IC)=0.	00016110
	A2(IC)=0.	00016120
	THINC2=THINC/2.	00016130
	THINC2=(KI-1)*THINC+CR	00016140
	XB(IC)=AX*DCOS(TH)	00016150

```

YB(IC)=AX*DSIN(TH)
THTA(IC)=TH/RAD
IF(J.GT.4) GO TO 20
IF(NELTYP(I).NE.2) GO TO 40
IF(J.EQ.1.AND.(K.EQ.1.OR.K.EQ.4)) A1(IC)=
  X-BSTR*DABS(DCOS(TH))
IF(J.EQ.2.AND.(K.EQ.3.OR.K.EQ.4)) A1(IC)=
  X-BSTR*DABS(DSIN(TH))
IF(J.EQ.3.AND.(K.EQ.2.OR.K.EQ.3)) A1(IC)=
  X-BSTR*DABS(DCOS(TH))
IF(J.EQ.4.AND.(K.EQ.1.OR.K.EQ.2)) A1(IC)=
  X-BSTR*DABS(DSIN(TH))
40 CONTINUE
20 CONTINUE
10 CONTINUE
RETURN
END

```

C 600

SUBROUTINE INFLN(WGHT,H,HRNK,J,KJ,I,NOPT)

```

IMPLICIT REAL*8(A-H,O-Z)
DIMENSION ELSTFF(50,10,10),WGHT(500),WK(150)
DIMENSION ELSTS(50,50,10),STSV(50),STSA(50,10)
DIMENSION AN(10,7),UVOUT(20)
DIMENSION PHI(3,7,400),STEMP(10,10),AO(10,3)
DIMENSION FINF(10,10),SINF(10,10),AINV(3,3)
DIMENSION APSX(500),APSY(500),APSXY(500)
DIMENSION H(2),XC(5),YC(5),NPLY(2)
DIMENSION IC(10)
DIMENSION A(10,10),ATEMP2(10,10)
COMMON/UV/UVOUT
COMMON/XCYC/XC,YC
COMMON/ELP/AX,BX,NOUT,NSTS
COMMON/ELS/ELSTFF,ELSTSS
COMMON/STS/STSV
COMMON/INF1/APSX,APSY,APSXY
COMMON/IYP/NPLY
COMMON/INV/AINV

```

C 600

COMPUTE ELEMENT STIFFNESS COEFFICIENTS

```

IF(J.GT.1) GO TO 200
DO 7 III=1,10
7 IC(III)=III
DO 100 IN1=1,3
DO 100 IN2=1,7
DO 100 IN3=1,400
100 PHI(IN1,IN2,IN3)=0.000
DO 444 N1=1,10
AO(N1,1)=0.000
AO(N1,2)=0.000
AO(N1,3)=0.000
DO 444 N2=1,7
444 AN(N1,N2)=0.000
DO 110 IN1=1,10
DO 110 IN2=1,10
SINF(IN1,IN2)=0.000
STEMP(IN1,IN2)=0.000

```

00016160
00016170
00016180
00016190
00016200
00016210
00016220
00016230
00016240
00016250
00016260
00016270
00016280
00016290
00016300
00016310
00016320
00016330
00016340
00016350
00016360
00016370
00016380
00016390
00016400
00016410
00016420
00016430
00016440
00016450
00016460
00016470
00016480
00016490
00016500
00016510
00016520
00016530
00016540
00016550
00016560
00016570
00016580
00016590
00016600
00016610
00016620
00016630
00016640
00016650
00016660
00016670
00016680
00016690
00016700
00016710
00016720
00016730
00016740
00016750

```

110 FINF(IN1,IN2)=0.000
200 CONTINUE
C
C STRESSES AND DISPLACEMENTS ARE STORED
C FOR EACH LOAD CASE
DO 2107 KLK=1,8
2107 AN(KLK,J)=UVOUT(KLK)
IF(NRNK.EQ.5) GO TO 2221
IF(J.EQ.1) UVOUT(5)=UVOUT(9)
IF(J.EQ.1) UVOUT(7)=UVOUT(9)
IF(J.EQ.2) UVOUT(2)=UVOUT(16)
IF(J.EQ.2) UVOUT(8)=UVOUT(16)
IF(J.EQ.3) UVOUT(1)=UVOUT(13)
IF(J.EQ.3) UVOUT(3)=UVOUT(16)
IF(J.EQ.4) UVOUT(4)=UVOUT(12)
IF(J.EQ.4) UVOUT(6)=UVOUT(12)
2221 CONTINUE
IF(NRNK.EQ.7 AND J.LT.5) GO TO 371
AN(9,J)=(UVOUT(9)+UVOUT(13))/2.
AN(10,J)=(UVOUT(12)+UVOUT(16))/2.
GO TO 372
371 AN(9,J)=UVOUT(7+2*J)
AN(10,J)=UVOUT(8+2*J)
372 CONTINUE
DO 15 IS=1,NSTS
15 STSA(IS,J)=STGV(IS)
DO 10 IS=1,NGPT
PHI(1,J,IS)=APSX(IS)
10 CONTINUE
DO 20 IS=1,NGPT
PHI(2,J,IS)=APSY(IS)
20 CONTINUE
DO 30 IS=1,NGPT
PHI(3,J,IS)=APSYX(IS)
30 CONTINUE
IF(J.LT.NRNK) RETURN

INTEGRATION OF STRESSES
DO 1010 III=1,10
1010 CONTINUE
NTR=NRNK+3
DO 45 IK=1,NTR
DO 45 JK=1,NTR
45 FINF(IK,JK)=0.
H1=H/VJ)*NPLY(KJ)
DO 50 LI=1,NGPT
DO 50 LJ=1,3
DO 50 KI=1,NRNK
SUM=0.
DO 70 IL=1,3
SUM=SUM+H1*AINV(LI,IL)*PHI(IL,KI,LJ)
70 CONTINUE
STEMP(LJ,KI)=SUM
60 CONTINUE
DO 80 LK=1,NRNK
DO 80 LJ=1,NRNK
SUM=0.
DO 90 IL=1,3

```

```

00016760
00016770
00016780
00016790
00016800
00016810
00016820
00016830
00016840
00016850
00016860
00016870
00016880
00016890
00016900
00016910
00016920
00016930
00016940
00016950
00016960
00016970
00016980
00016990
00017000
00017010
00017020
00017030
00017040
00017050
00017060
00017070
00017080
00017090
00017100
00017110
00017120
00017130
00017140
00017150
00017160
00017170
00017180
00017190
00017200
00017210
00017220
00017230
00017240
00017250
00017260
00017270
00017280
00017290
00017300
00017310
00017320
00017330
00017340
00017350

```


SUM=SUM+PHI(IL,LK,LI)*STEMP(IL,LJ)	00017360
90 CONTINUE	00017370
FINF(LK,LJ)=FINF(LK,LJ)+SUM*WGH(LI)	00017380
80 CONTINUE	00017390
50 CONTINUE	00017400
DO 51 III=1, NRNK	00017410
DO 51 JJJ=1, NRNK	00017420
51 STMP(III, JJJ)=(FINF(III, JJJ)+FINF(JJJ, III))/2.0D0	00017430
DO 52 III=1, NRNK	00017440
DO 52 JJJ=1, NRNK	00017450
52 FINF(III, JJJ)=STEMP(III, JJJ)	00017460
CALL LINV2F(FINF, N, NRK, 10, SINF, 4, WK, IER)	00017470
DO 410 IA=1, NTR	00017480
AO(IA, 1)=0.5+0.5*(-1.0)**(IA+1)	00017490
AO(IA, 2)=0.5+0.5*(-1.0)**(IA)	00017500
410 AO(IA, 3)=0.0D0	00017510
AO(1, 3)=DABS(YC(4))	00017520
AO(2, 3)=-DABS(XC(4))	00017530
AO(3, 3)=-DABS(YC(3))	00017540
AO(4, 3)=-DABS(XC(3))	00017550
AO(5, 3)=-DABS(YC(2))	00017560
AO(6, 3)=DABS(XC(2))	00017570
AO(7, 3)=DABS(YC(1))	00017580
AO(8, 3)=DABS(XC(1))	00017590
DO 420 KK=1, NTR	00017600
DO 420 LL=1, NRNK	00017610
SUM=0.0D0	00017620
DO 430 JJ=1, NRNK	00017630
430 SUM=SUM+AN(KK, JJ)*SINF(JJ, LL)	00017640
420 STMP(KK, LL)=SUM	00017650
DO 440 KK=1, NTR	00017660
N1=NTR-2	00017670
DO 440 JJ=N1, NTR	00017680
440 STMP(KK, JJ)=AO(KK, JJ-NRNK)	00017690
CALL LINV2F(STMP, NTR, 10, FINF, 4, WK, IER)	00017700
DO 450 II=1, NRNK	00017710
DO 450 JJ=1, NTR	00017720
SUM=0.0D0	00017730
DO 460 KK=1, NRNK	00017740
460 SUM=SUM+SINF(II, KK)*FINF(KK, JJ)	00017750
450 STMP(II, JJ)=SUM	00017760
DO 470 II=1, NTR	00017770
DO 470 JJ=1, NTR	00017780
SUM=0.0D0	00017790
DO 480 KK=1, NRNK	00017800
480 SUM=SUM+FINF(KK, II)*STEMP(KK, JJ)	00017810
ELSTFF(I, II, JJ)=SUM	00017820
A(II, JJ)=SUM	00017830
470 CONTINUE	00017840
DO 550 II=1, NRNK	00017850
DO 550 JJ=1, NTR	00017860
SUM=0.0D0	00017870
DO 560 KK=1, NRNK	00017880
560 SUM=SUM+SINF(II, KK)*FINF(KK, JJ)	00017890
550 STMP(II, JJ)=SUM	00017900
DO 570 II=1, NST3	00017910
DO 570 JJ=1, NTR	00017920
SUM=0.0D0	00017930
DO 580 KK=1, NRNK	00017940
580 SUM=SUM+A(II, KK)*STEMP(KK, JJ)	00017950

```

570 ELSTOP(1,II,1)=SUM
RETURN
END

```

C
C
C

```

SUBROUTINE SMAX(HT,KJ,1)
IMPLICIT REAL*8(A-H,O-Z)
DIMENSION AINV(3,3),STULT(3,2),AVN(3)
DIMENSION NV(3)
DIMENSION PSMX(50,4),STM(3),CM(2)
DIMENSION NPLY(2),NUMPLY(2),ANG(5,2),IPLY(100,2)
DIMENSION E1(2),E2(2),Q12(2),V12(2),V21(2)
COMMON/MD/E1,E2,Q12,V12,V21
COMMON/LYP/NPLY,NUMPLY,ANG,IPLY
COMMON/STMT/STM,CM
COMMON/INV/AINV
COMMON/SMX/PSMX
COMMON/STN/STULT
DATA CM/0.0/
IF(CM(KJ).EQ.0) GO TO 222
PSMX(1,1)=STM(1)
PSMX(1,2)=STM(2)
PSMX(1,3)=STM(3)
PSMX(1,4)=STM(2)
RETURN

```

222 CONTINUE

C
C
C

```

COMPUTE LAMINATE FAILURE LOADS BASED ON MAXIMUM
FIBER STRAINS FOR EACH FAILURE MODE

```

```

DO 100 K=1,3
DO 10 II=1,3
NV(II)=0
10 AVN(II)=0.000
IF(K.EQ.1) NV(1)=1
IF(K.EQ.2) NV(1)=-1
IF(K.EQ.3) NV(3)=1
DO 15 II=1,3
DO 15 JJ=1,3
AVN(II)=AVN(II)+AINV(II,JJ)*NV(JJ)
15 CONTINUE
NP=NUMPLY(KJ)
SMX=0.000
RAD=DARCOS(-1.000)/180.00
DO 25 II=1,NP
TH=ANG(II,KJ)*RAD
E1=DCOS(TH)*2*AVN(1)+AVN(2)*DSIN(TH)*2+
*DCOS(TH)*DSIN(TH)*AVN(3)
IF(K.EQ.1) GO TO 65
EPRT=E1/STULT(2,KJ)
GO TO 50
65 IF(K.EQ.2) GO TO 75
EPRT=E1/STULT(1,KJ)
GO TO 50
75 EPRT=E1/STULT(2,KJ)
50 CONTINUE
IF(DABS(SMX).LT.DABS(EPRT)) SMX=EPRT
25 CONTINUE
IF(DABS(SMX).GT.1.00-10) GO TO 555

```

```

00017960
00017970
00017980
00017990
00018000
00018010
00018020
00018030
00018040
00018050
00018060
00018070
00018080
00018090
00018100
00018110
00018120
00018130
00018140
00018150
00018160
00018170
00018180
00018190
00018200
00018210
00018220
00018230
00018240
00018250
00018260
00018270
00018280
00018290
00018300
00018310
00018320
00018330
00018340
00018350
00018360
00018370
00018380
00018390
00018400
00018410
00018420
00018430
00018440
00018450
00018460
00018470
00018480
00018490
00018500
00018510
00018520
00018530
00018540
00018550

```

	PSMX(I,K)=STULT(3,KJ)*Q12(KJ)	00018560
	GO TO 100	00018570
555	CONTINUE	00018580
	PSMX(I,K)=DABS(1.0D0/SMX)	00018590
100	CONTINUE	00018600
	PSMX(I,4)=PSMX(1,2)	00018610
	RETURN	00018620
	END	00018630
C		00018640
C		00018650
C		00018660
	SUBROUTINE POLY(W,AST,J,K,NCOL,LTNCH)	00018670
		00018680
		00018690
	IMPLICIT REAL*8(A-H,O-Z)	00018700
	DIMENSION XC(5),YC(5),A1(200),A2(200),XB(200)	00018710
	DIMENSION YB(200),T(200),A1A(4),A2A(4)	00018720
	COMMON/CHT1/XB,YB,A1,A2,T	00018730
	COMMON/XCYC/XC,YC	00018740
C		00018750
C	ARRAY COLLOCATION POINTS AROUND EXTERIOR	00018760
C	BOUNDARY AND APPLY STRESS BOUNDARY	00018770
C	CONDITION	00018780
		00018790
	DO 120 I=1,4	00018800
	A1A(I)=0.0	00018810
	A2A(I)=0.0	00018820
120	CONTINUE	00018830
	IF(LTNCH.EQ.1) A1A(I)=AST	00018840
	IF(LTNCH.EQ.2) A1A(I)=-AST	00018850
	J=0	00018860
	XC(5)=XC(1)	00018870
	YC(5)=YC(1)	00018880
	DO 10 I=1,4	00018890
	X=XC(I)-XC(I+1)	00018900
	Y=YC(I+1)-YC(I)	00018910
	IF(X.EQ.0.) X=1.D-6	00018920
	IF(Y.EQ.0.) Y=1.D-6	00018930
	TH=DATAH2(X,Y)	00018940
	TH=TH*180./DARCOS(-0.1D1)	00018950
	DX=(XC(I+1)-XC(I))/(NCOL+1)	00018960
	DY=(YC(I+1)-YC(I))/(NCOL+1)	00018970
	DO 20 II=1,NCOL	00018980
	J=J+1	00018990
	IF(I.EQ.1.OR.I.EQ.3) GO TO 23	00019000
	YB(J)=YC(I)	00019010
	XB(J)=XC(I)+DX*(II+.5)	00019020
	IF(II.EQ.1) XB(J)=XC(I)+(DX/2.)	00019030
	GO TO 24	00019040
23	CONTINUE	00019050
	YB(J)=YC(I)+DY*(II+.5)	00019060
	IF(II.EQ.1) YB(J)=YC(I)+(DY/2.)	00019070
	XB(J)=XC(I)	00019080
24	T(J)=TH	00019090
	A1(J)=A1A(I)	00019100
	A2(J)=A2A(I)	00019110
20	CONTINUE	00019120
10	CONTINUE	00019130
	RETURN	00019140
	END	00019150

SUBROUTINE CIRC(H,AST,JK,K,LTNGM)

ARRAY COLLOCATION POINTS AROUND INNER BOUNDARY AND
APPLY BEARING LOAD IN A COSINUSOIDAL DISTRIBUTION

```

IMPLICIT REAL8(A-H,O-Z)
DIMENSION X(600),Y(600),THTA(200),A1(200),A2(200)
DIMENSION XB(200),YB(200)
COMMON/FB1/BSTR,XSTR
COMMON/CMT1/XB,YB,A1,A2,THTA
COMMON/CMT2/X,Y
COMMON/ELP/A,B,N

```

```
CON=-1.0
XSTR=AST
BSTR=(2.*XSTR)/(DARCOS(CON)*NB)
```

14MG 21-4

DO 20 I=1,N

$$TH = ((I-1) \times 2 + 1) \times DARCOS(CON) / N$$
$$X(t) = A \sin \omega t \cos(\theta t)$$

```
Y(I)=BMDSIN(TH)
CS=-X(I)*RMB/(Y(I)*(A+A))
```

```
IF(Y(I).GT.0)THTA(I)=DATAN(CS)-DARCOS(CON)/2.
IF(Y(I).LT.0)THTA(I)=DATAN(CS)+DARCOS(CON)/2.
```

```

THTA(JK)=THT1(JK)+180./DARCOS(CON)
IF(LYNCH.EQ.2) GO TO 25

```

```
IF(I.LT.(N-1)).AND.I.LT.(N-N2)) GO TO 204
GO TO 30
```

```
25 IF(I.LE.(NQ+2).OR.I.GE.(N-NQ-1)) GO TO 204
```

30 CONTINUE

ALCJX) = 0.

$$A_2(1, k) = 0.$$
$$YB(14) = X(1)$$

Y:1 5 11 1 2 4 (1)

2720

204

• 3577 •

1997-1998

$$g(y) = x(y)$$

13() 2Y

CONCLUSION

RETURN

• 10

SUBROUTINE FIGECM(M,PHS,KJ,NOPT4,NCLL)

INTERNET FORUM: PRIVATE SECURITY ANALYSIS BY A
 COLLECTIVE COLLABORATION TECHNOLOGY

00019160
00019170
00019180
00019190
00019200
00019210
00019220
00019230
00019240
00019250
00019260
00019270
00019280
00019290
00019300
00019310
00019320
00019330
00019340
00019350
00019360
00019370
00019380
00019390
00019400
00019410
00019420
00019430
00019440
00019450
00019460
00019470
00019480
00019490
00019500
00019510
00019520
00019530
00019540
00019550
00019560
00019570
00019580
00019590
00019600
00019610
00019620
00019630
00019640
00019650
00019660
00019670
00019680
00019690
00019700
00019710
00019720
00019730
00019740
00019750

	IMPLICIT REAL*8(A-H,O-Z)	00019760
	DIMENSION A(3,3),WK(25),AI(3,3),AZ(5),WKK(121),BC(300)	00019770
	DIMENSION CH(4),H(2)	00019780
	COMPLEX*16 ORHS(100)	00019790
	COMPLEX*16 CM(300,90),CMC(300,90),CMCTCM(90,90),RHS(90)	00019800
	COMMON/ROOTS/R1,R2	00019810
	COMMON/AMT/A	00019820
	COMMON/TERMS/P1,Q1,P2,Q2	00019830
	COMMON/ELP/AX,BX,NGUT,NSTS	00019840
	COMMON/SER/NT,NB	00019850
	COMMON/INV/AI	00019860
	COMPLEX*16 Z(4),Z1,Z2,Q1,Q2,P1,P2,R1,R2,WA(14883)	00019870
C		00019880
C	AMATRX CALCULATES THE LAMINATE 'A' MATRIX	00019890
C		00019900
	CALL AMATRX(H,PHS,KJ)	00019910
	N=3	00019920
	IDGT=4	00019930
	IA=3	00019940
C		00019950
C	LINV2F INVERTS THE 'A' MATRIX	00019960
C		00019970
	CALL LINV2F(A,N,IA,AI,IDGT,WK,IER)	00019980
	NDEG=4	00019990
	AZ(1)=AI(1,1)/AI(2,2)	00020000
	AZ(2)=-2.*AI(1,3)/AI(2,2)	00020010
	AZ(3)=(2.*AI(1,2)+AI(3,3))/AI(2,2)	00020020
	AZ(4)=-2.*AI(2,3)/AI(2,2)	00020030
	AZ(5)=1.0D0	00020040
C		00020050
C	ZRPOLY FINDS THE ROOTS OF THE CHARACTERISTIC EQUATION	00020060
C		00020070
C	CALL ZRPOLY(AZ,NDEG,Z,IER)	00020080
C		00020090
C	Z(2) AND Z(4) ARE THE COMPLEX CONJUGATES OF Z(1)	00020100
C	AND Z(3) RESPECTIVELY	00020110
C		00020120
	R1=Z(1)	00020130
	R2=Z(3)	00020140
		00020150
C	THE TWO ROOTS MUST BE CHECKED FOR A UNITARY COMPONENT	00020160
C	IN EITHER THE REAL OR IMAGINARY PART; SUCH AN	00020170
C	OCCURRENCE SIGNIFIES A QUASI-ISOTROPIC LAYUP AND	00020180
C	THE VALUE MUST BE PERTURBED SLIGHTLY IN ORDER TO	00020190
C	AVOID A SINGULAR MATRIX	00020200
		00020210
	CH(1)=R1	00020220
	CH(2)=(0.0,-1.0)*R1	00020230
	CH(3)=R2	00020240
	CH(4)=(0.0,-1.0)*R2	00020250
	DO 30 IJK=1,4	00020260
	AR=DABS(CH(IJK))	00020270
	IF(AR.LE 1.0) GO TO 31	00020280
	GO TO 32	00020290
31	IF((1.0-AR).LT.0.02) CH(IJK)=0.98	00020300
	GO TO 30	00020310
32	IF((AR-1.0).LT.0.02) CH(IJK)=1.02	00020320
30	CONTINUE	00020330
	R1=DCMPLX(CH(1),CH(2))	00020340
	R2=DCMPLX(CH(3),CH(4))	00020350

C
C
C

CONSTANTS P1,P2,Q1,Q2 ARE NEEDED FOR STRESS CALCULATIONS

P1=AI(1,1)*R1**2+AI(1,2)*AI(1,3)*R1
P2=AI(1,1)*R2**2+AI(1,2)*AI(1,3)*R2
Q1=AI(2,2)/R1+AI(1,2)*R1-AI(2,3)
Q2=AI(2,2)/R2+AI(1,2)*R2-AI(2,3)

C
C
C

INPUTS AIN1(I),AIN2(I) ETC. REFER TO BOUNDARY CONDITIONS

NT4=4*NT
NT8=8*NT
NT8P4=8*NT+4
NT8P2=8*NT+2
NT8P1=8*NT+1
NB2=2*NB
NWK=NT8P1*(NT8P1+2)
CALL CMAT(BC,CMCTCM,CMC,CM,RHS,GRHS,NT4,NT8,NT8P4,NT8P2,
NT8P1,NB2,NWK,WA,NKK,NOPT4,KJ,NCLL)
RETURN
END

C
C
C
C
C
C

SUBROUTINE AMATRX(H,PHS,K)

ASSEMBLE THE A MATRIX

10 IMPLICIT REAL*8(A-H,O-Z)
DIMENSION A(3,3),ANO(5,2),H(2),NPLY(2),NUMPLY(2)
DIMENSION E1(2),E2(2),Q12(2),V12(2),V21(2)
DIMENSION IPLY(100,2)
COMMON/MD/E1,E2,Q12,V12,V21
COMMON/AMT/A
COMMON/LYP/NPLY,NUMPLY,ANO,IPLY
THKNES=NPLY(K)*H(K)
DENO=1.-E2(K)*V12(K)*H2/E1(K)
Q11=E1(K)/DENO
Q22=E2(K)/DENO
Q12=V12(K)*Q22
Q21=Q12
Q33=Q12(K)
DO 10 I=1,3
DO 10 J=1,3
10 A(I,J)=0.0D0
NN=NPLY(K)
T=H(K)
DO 20 I=1,NN
LP=IPLY(I,K)
THTAI=(ANO(LP,K)+PHS)*DARCOS(-1.D0)/180.D0
C=DCOS(THTAI)
S=DSIN(THTAI)
A(1,1)=(Q11*CNH4+2.*(Q12+2.*Q33)*CNH3*SHS+Q22*SHH4)*T+A(1,1)
A(2,2)=(Q11*SHH4+2.*(Q12+2.*Q33)*CNH3*SHS+Q22*CNH4)*T+A(2,2)
A(1,2)=((Q11+Q22-4.*Q33)*CNH3*SHS+Q12*(CNH4+SHH4))*T+A(1,2)
A(2,1)=A(1,2)
A(3,3)=((Q11+Q22-2.*Q12-2.*Q33)*CNH3*SHS+Q33*(CNH4+SHH4))*T+A(3,3)
A(1,3)=((Q11-Q12-2.*Q33)*CNH3*SHS+(Q12-Q22+2.*Q33)*SHH3*CN)*T+A(1,3)
A(2,3)=((Q11-Q12-2.*Q33)*SHH3*CN+(Q12-Q22+2.*Q33)*CNH3*SHS)*T+A(2,3)
A(3,2)=A(2,3)

00020360
00020370
00020380
00020390
00020400
00020410
00020420
00020430
00020440
00020450
00020460
00020470
00020480
00020490
00020500
00020510
00020520
00020530
00020540
00020550
00020560
00020570
00020580
00020590
00020600
00020610
00020620
00020630
00020640
00020650
00020660
00020670
00020680
00020690
00020700
00020710
00020720
00020730
00020740
00020750
00020760
00020770
00020780
00020790
00020800
00020810
00020820
00020830
00020840
00020850
00020860
00020870
00020880
00020890
00020900
00020910
00020920
00020930
00020940
00020950

<pre> 20 A(3,1)=A(1,3) CONTINUE DO 53 I=1,3 DO 53 J=1,3 A(I,J)=A(I,J)/THKNES 53 CONTINUE RETURN END </pre>	<pre> 00020960 00020970 00020980 00020990 00021000 00021010 00021020 00021030 00021040 00021050 00021060 00021070 00021080 00021090 00021100 00021110 00021120 00021130 00021140 00021150 00021160 00021170 00021180 00021190 00021200 00021210 00021220 00021230 00021240 00021250 00021260 00021270 00021280 00021290 00021300 00021310 00021320 00021330 00021340 00021350 00021360 00021370 00021380 00021390 00021400 00021410 00021420 00021430 00021440 00021450 00021460 00021470 00021480 00021490 00021500 00021510 00021520 00021530 00021540 00021550 </pre>
<pre> SUBROUTINE CMAT(BC,CMCTCM,CMC,CM,RHS,GRHS,NT4,NT8,NT8P4,NT8P2, 1NT8P1,NB2,NWK,WA,WKK,NOPT4,KJ,NCOL) </pre>	
<pre> CMAT OUTPUTS STRESSES, STRAINS, AND DISPLACEMENTS AT SPECIFIED COORDINATES </pre>	
<pre> IMPLICIT REAL*8(A-H,O-Z) DIMENSION ASX(400),ASXY(400),UVOUT(20) DIMENSION XC(5),YC(5) DIMENSION THTA(200),X(200),Y(200),AMAT(3,3) DIMENSION AIN1(200),AIN2(200),BC(NB2) DIMENSION WKK(NT8P1),WORK(700) DIMENSION XOUT(300),YOUT(600),STSV(50) DIMENSION FUR(400),FTHT(400),FSMR(400) DIMENSION RTHT(400),REPX(400),REPY(400),REPLY(400) DIMENSION APSX(500),APSY(500),APSXY(500) COMPLEX*16 CMCTCM(NT8P1,NT8P1),RHS(NT8P1),PHI1D,PHI2D,XETA1,XETA2 COMPLEX*16 ACD(25,25),ACD2(25,25),RHS2(25) COMPLEX*16 UO,VO COMPLEX*16 CM(NB2,NT8P4),CMC(NB2,NT8P1),Z1,Z2,Z11,Z22,R1,R2 COMPLEX*16 T11,T12,T21,T22,P11,P12,P21,P22 COMPLEX*16 P1,P2,Q1,Q2,DCMPLX,CO,CSUM,GRHS(NT8P2) COMPLEX*16 PHI1DP,PHI2DP,PHI1DN,PHI2DN COMPLEX*16 PHI1P,PHI2P,PHI1N,PHI2N,PHI1,PHI2 COMPLEX*16 PHI3N,PHI3P,PHI3,PHI4N,PHI4P,PHI4 COMPLEX*16 SV11,SV12,SV21,SV22,RB11,RB21,RB1B,RB2B COMPLEX*16 R1B,R2B,P1B,P2B,Q1B,Q2B,WA(NWK) COMMON/INF1/APSX,APSY,APSXY COMMON/XCYC/XC,YC COMMON/NCST/NCASE,NTYPE COMMON/XXY1/ASX,ASXY COMMON/STS/STSV COMMON/UV/UVOUT COMMON/ROOTS/R1,R2 COMMON/TERMS/P1,Q1,P2,Q2 COMMON/CMT1/X,Y,AIN1,AIN2,THTA COMMON/CMT2/XOUT,YOUT COMMON/FB2/FUR,FTHT,FSMR COMMON/CMT/RTHT,REPX,REPY,REPLY COMMON/ELP/AX,BX,HOUT,NSTS COMMON/SER/NT,NB COMMON/INV/AMAT IF(NOPT4.EQ.5.AND.NCASE.GT.1) GO TO 3335 DO 6666 I1=1,NT8P1 DO 6666 I2=1,NT8P1 6666 CMCTCM(I1,I2)=(0.0D0,U.0D0) A=AX </pre>	

```

B=BX
CO=(0.0,1.0)
RB11=(Q1-P1*R1)/(A-CO*R1*B)
RB21=(Q2-P2*R2)/(A-CO*R2*B)
REALR1=R1
REALR2=R2
REALP1=P1
REALP2=P2
REALQ1=Q1
REALQ2=Q2
RRB11=RB11
RRB21=RB21
AIMGR1=CO*R1
AIMGR2=CO*R2
AIMGP1=CO*P1
AIMGP2=CO*P2
AIMGQ1=CO*Q1
AIMGQ2=CO*Q2
ARB11=CO*RB11
ARB21=CO*RB21
R1B=DCMPLX(REALR1,AIMGR1)
R2B=DCMPLX(REALR2,AIMGR2)
P1B=DCMPLX(REALP1,AIMGP1)
P2B=DCMPLX(REALP2,AIMGP2)
Q1B=DCMPLX(REALQ1,AIMGQ1)
Q2B=DCMPLX(REALQ2,AIMGQ2)
RB11B=DCMPLX(RRB11,ARB11)
RB21B=DCMPLX(RRB21,ARB21)
JJJ=0
DO 1000 I=1,NB
J=I*2
THTAI=THTA(I)*DARCO5(-1.000)/180.00
C=DCOS(THTAI)
S=DSIN(THTAI)
P11=C*P1+S*Q1
P12=C*P2+S*Q2
P21=-S*P1+C*Q1
P22=-S*P2+C*Q2
T11=(C*C*R1/R1+S*S-2.*C*S*R1)
T12=(C*C*R2/R2+S*S-2.*C*S*R2)
T21=(-C*C*R1/R1+C*S-(C*C-S*S)*R1)
T22=(-C*C*R2/R2+C*S-(C*C-S*S)*R2)
Z1=X(I)+R1*Y(I)
Z2=X(I)+R2*Y(I)
Z11=CDSORT(Z1*Z1-A*A-R1*R1*B*B)
Z22=CDSORT(Z2*Z2-A*A-R2*R2*B*B)
REAL1=Z11
AIMG1=-CO*Z11
IF(DABS(REAL1).LE.1.D-14)REAL1=0.000
IF(DABS(AIMG1).LE.1.D-16)AIMG1=0.000
Z11=DCMPLX(REAL1,AIMG1)
REAL2=Z22
AIMG2=-CO*Z22
IF(DABS(REAL2).LE.1.D-16)REAL2=0.000
IF(DABS(AIMG2).LE.1.D-16)AIMG2=0.000
Z22=DCMPLX(REAL2,AIMG2)
XETA1=(Z1+Z11)/(A-CO*R1*B)
IF(CDABS(XETA1).LT.0.999) GO TO 300
GO TO 310
300 Z11=-Z11

```

```

00021560
00021570
00021580
00021590
00021600
00021610
00021620
00021630
00021640
00021650
00021660
00021670
00021680
00021690
00021700
00021710
00021720
00021730
00021740
00021750
00021760
00021770
00021780
00021790
00021800
00021810
00021820
00021830
00021840
00021850
00021860
00021870
00021880
00021890
00021900
00021910
00021920
00021930
00021940
00021950
00021960
00021970
00021980
00021990
00022000
00022010
00022020
00022030
00022040
00022050
00022060
00022070
00022080
00022090
00022100
00022110
00022120
00022130
00022140
00022150

```


	XETA1=(Z1+Z11)/(A-COXR1*B)	00022160
310	XETA2=(Z2+Z22)/(A-COXR2*B)	00022170
	IF(CDABS(XETA2).LT.0.999) GO TO 320	00022180
	GO TO 330	00022190
320	Z22=-Z22	00022200
	XETA2=(Z2+Z22)/(A-COXR2*B)	00022210
330	CONTINUE	00022220
	JJJ=JJJ+1	00022230
C		00022240
C	NORMAL & TANGENTIAL STRESS BOUNDARY CONDITIONS ARE IMPOSED	00022250
	DO 5 N=1,NT	00022260
	NP=N	00022270
	CM(J-1,N)=NP*XETA1*NNPNT11/Z11	00022280
	CM(J-1,2*NT+N)=NP*XETA2*NNPNT12/Z22	00022290
	CM(J,N)=NP*XETA1*NNPNT21/Z11	00022300
	CM(J,2*NT+N)=NP*XETA2*NNPNT22/Z22	00022310
	NN=-N	00022320
	CM(J-1,NT+N)=NN*XETA1*NNNNT11/Z11	00022330
	CM(J-1,3*NT+N)=NN*XETA2*NNNNT12/Z22	00022340
	CM(J,NT+N)=NN*XETA1*NNNNT21/Z11	00022350
	CM(J,3*NT+N)=NN*XETA2*NNNNT22/Z22	00022360
5	CONTINUE	00022370
	CM(J-1,NT8+1)=T11/Z11	00022380
	CM(J-1,NT8+2)=T12/Z22	00022390
	CM(J,NT8+1)=T21/Z11	00022400
	CM(J,NT8+2)=T22/Z22	00022410
1000	CONTINUE	00022420
	DO 195 I=1,NB2	00022430
	DO 196 J=1,NT4	00022440
	REAL1=CM(I,J)	00022450
	AIMG1=-COXCM(I,J)	00022460
	IF(DABS(REAL1).LE.1.D-16)REAL1=0.0D0	00022470
	IF(DABS(AIMG1).LE.1.D-16)AIMG1=0.0D0	00022480
	CM(I,J)=DCMPLX(REAL1,AIMG1)	00022490
	AIMG2=-AIMG1	00022500
	CM(I,NT4+J)=DCMPLX(REAL1,AIMG2)	00022510
196	CONTINUE	00022520
195	CONTINUE	00022530
	DO 295 I=1,NB2	00022540
	DO 296 J=1,2	00022550
	REAL1=CM(I,NT8+J)	00022560
	AIMG1=-COXCM(I,NT8+J)	00022570
	IF(DABS(REAL1).LE.1.D-16)REAL1=0.0D0	00022580
	IF(DABS(AIMG1).LE.1.D-16)AIMG1=0.0D0	00022590
	CM(I,NT8+J)=DCMPLX(REAL1,AIMG1)	00022600
	AIMG2=-AIMG1	00022610
	CM(I,NT8+2+J)=DCMPLX(REAL1,AIMG2)	00022620
296	CONTINUE	00022630
295	CONTINUE	00022640
	SV11=(P2*Q1B-Q2*Q1B)/(Q1*P2-Q2*P1)	00022650
	SV12=(P2*Q2B-Q2*P2B)/(Q1*P2-Q2*P1)	00022660
	SV21=(Q1*P1B-Q1*P1B)/(Q1*P2-Q2*P1)	00022670
	SV22=(Q1*P2B-Q2*P1B)/(Q1*P2-Q2*P1)	00022680
	DO 139 I=1,NB2	00022690
		00022700
C		00022710
C	IMPOSE RIGID BODY ROTATION CONDITION	00022720
	CM(I,2*NT+1)=-CM(I,1)*RB21/RB11+CM(I,2*NT+1)	00022730
	CM(I,4*NT+1)=-CM(I,1)*RB11B/RB11+CM(I,4*NT+1)	00022740
		00022750

```

COC
CM(I,6*NT+1)=-CM(I,1)*RB21B/R511+CM(I,6*NT+1)
CM(I,1)=(0.0,0.0)
IMPOSE SINGLE-VALUEDNESS CONDITION
CM(I,NT8+3)=CM(I,NT8+1)*SV11+CM(I,NT8+3)
CM(I,NT8+4)=CM(I,NT8+1)*SV12+CM(I,NT8+4)
CM(I,NT8+3)=CM(I,NT8+2)*SV21+CM(I,NT8+3)
CM(I,NT8+4)=CM(I,NT8+2)*SV22+CM(I,NT8+4)
CM(I,NT8+1)=(0.0,0.0)
CM(I,NT8+2)=(0.0,0.0)
139 CONTINUE
DO 141 I=1,NB2
DO 142 J=2,NT8
142 CM(I,J-1)=CM(I,J)
CM(I,NT8)=CM(I,NT8+3)
CM(I,NT8+1)=CM(I,NT8+4)
141 CONTINUE
DO 95 I=1,NB2
DO 96 J=1,NT8P1
REAL1=CM(I,J)
AIM01=-CM(I,J)
IF(DABS(REAL1).LE.1.D-16)REAL1=0.0D0
IF(DABS(AIM01).LE.1.D-16)AIM01=0.0D0
CM(I,J)=DCMPLX(REAL1,AIM01)
AIM02=-AIM01
CMC(I,J)=DCMPLX(REAL1,AIM02)
96 CONTINUE
95 CONTINUE
DO 100 I=1,NT8P1
DO 100 J=1,NT8P1
CSUM=(0.0,0.0)
DO 110 K=1,NB2
110 CSUM=CMC(K,I)*CM(K,J)+CSUM
CMCTCM(I,J)=CSUM
100 CONTINUE
3335 CONTINUE
DO 120 I=1,NB
J=I+2
BC(J-1)=AIN1(I)
120 BC(J)=AIN2(I)
DO 130 I=1,NT8P1
CSUM=(0.0,0.0)
DO 140 K=1,NB2
140 CSUM=CMC(K,I)*BC(K)+CSUM
130 RHS(I)=CSUM
IJOB=0
IF(NOPY4.EQ.5.AND.NCASE.GT.1) IJGB=2
M=1
CALL LEQ2C(CMCTCM,NT8P1,NT8P1,RHS,M,NT8P1,IJOB,WA,WKK,IER)
IF(IER.EQ.129) WRITE(6,11)
11 FORMAT(' TERMINAL ERROR(CMCTCM), IER = 129')
GRHS(1)=-(RHS(2*NT)*RB21+RHS(4*NT)*RB11B+RHS(6*NT)*RB21B)/RB11
GRHS(8*NT+1)=RHS(8*NT)*SV11+RHS(8*NT+1)*SV12
GRHS(8*NT+2)=RHS(8*NT)*SV21+RHS(8*NT+1)*SV22
DO 151 I=2,NT8
151 GRHS(I)=RHS(I-1)
C
C
C STRESS AND STRAIN CALCULATION

```

```

00022760
00022770
00022780
00022790
00022800
00022810
00022820
00022830
00022840
00022850
00022860
00022870
00022880
00022890
00022900
00022910
00022920
00022930
00022940
00022950
00022960
00022970
00022980
00022990
00023000
00023010
00023020
00023030
00023040
00023050
00023060
00023070
00023080
00023090
00023100
00023110
00023120
00023130
00023140
00023150
00023160
00023170
00023180
00023190
00023200
00023210
00023220
00023230
00023240
00023250
00023260
00023270
00023280
00023290
00023300
00023310
00023320
00023330
00023340
00023350

```

```

RAD=DARCOS(-1.000)/180.00
IC=1
IC2=1
SUMU1=0.000
SUMV1=0.000
SUMU2=0.000
SUMV2=0.000
NADD=0
IF(NOPT4.EQ.1) GO TO 1195
IF(NTYPE.NE.2.OR.NCASE.GT.4) GO TO 1196
NADD=4*NCUL
NIC=1
DO 197 II=1,NADD
  XOUT(NOUT+NSTS+II)=X(II)
197 YOUT(NOUT+NSTS+II)=Y(II)
  MN=NOUT+NSTS+NADD
  XOUT(MN+1)=AX
  YOUT(MN+1)=0.000
  XOUT(MN+2)=0.000
  YOUT(MN+2)=-AX
  XOUT(MN+3)=AX*DCOS(177.00*RAD)
  YOUT(MN+3)=AX*DSIN(177.00*RAD)
  XOUT(MN+4)=0.000
  YOUT(MN+4)=AX
  NADD=NADD+4
GO TO 1195
1196 CONTINUE
  NADD=8
  NIC=2
  MN=NOUT+NSTS
  DO 199 III=1,4
    ICM=5-III
    XOUT(MN+III)=XC(ICM)
    YOUT(MN+III)=YC(ICM)
199 CONTINUE
    MN=MN+4
    XOUT(MN+1)=AX
    YOUT(MN+1)=0.000
    XOUT(MN+2)=0.000
    YOUT(MN+2)=-AX
    XOUT(MN+3)=AX*DCOS(177.00*RAD)
    YOUT(MN+3)=AX*DSIN(177.00*RAD)
    XOUT(MN+4)=0.000
    YOUT(MN+4)=AX
1195 CONTINUE
  NRCF=NOUT
  IF(NOPT4.EQ.5) NRCF=NOUT+NSTS+NADD
  NINC=NSTS/4
  DO 190 K=1,NRCF
    Z1=XOUT(K)+R1*YOUT(K)
    Z2=XOUT(K)+R2*YOUT(K)
    Z11=CDSQRT(Z1*Z1-AXA-R1*NR1*MB)
    Z22=CDSQRT(Z2*Z2-AXA-R2*NR2*MB)
    XETA1=(Z1+Z11)/(A-CO*NR1*MB)
    IF(CDABS(XETA1).LT.0.999) GO TO 400
    GO TO 410
400 Z11=-Z11
    XETA1=(Z1+Z11)/(A-CO*NR1*MB)
410 XETA2=(Z2+Z22)/(A-CO*NR2*MB)
    IF(CDABS(XETA2).LT.0.999) GO TO 420

```

```

00023360
00023370
00023380
00023390
00023400
00023410
00023420
00023430
00023440
00023450
00023460
00023470
00023480
00023490
00023500
00023510
00023520
00023530
00023540
00023550
00023560
00023570
00023580
00023590
00023600
00023610
00023620
00023630
00023640
00023650
00023660
00023670
00023680
00023690
00023700
00023710
00023720
00023730
00023740
00023750
00023760
00023770
00023780
00023790
00023800
00023810
00023820
00023830
00023840
00023850
00023860
00023870
00023880
00023890
00023900
00023910
00023920
00023930
00023940
00023950

```

```

GO T) 430
420 Z22=-Z22
XETA1=(Z2+Z22)/(A-C0*R2*B)
430 CONTINUE
PHI1DP=(0.0,0.0)
PHI2DP=(0.0,0.0)
PHI1DN=(0.0,0.0)
PHI2DN=(0.0,0.0)
PHI1P=(0.0,0.0)
PHI2P=(0.0,0.0)
PHI1N=(0.0,0.0)
PHI2N=(0.0,0.0)
DO 170 N=1,NT
NP=N
NN=-N
PHI1DP=NP*XETA1*NN*GRHS(N)/Z11+PHI1DP
PHI1DN=NN*XETA1*NN*GRHS(N)/Z11+PHI1DN
PHI2DP=NP*XETA2*NN*GRHS(2*NT+N)/Z22+PHI2DP
PHI2DN=NN*XETA2*NN*GRHS(3*NT+N)/Z22+PHI2DN
PHI1P=XETA1*NN*GRHS(N)+PHI1P
PHI1N=XETA1*NN*GRHS(NT+N)+PHI1N
PHI2P=XETA2*NN*GRHS(2*NT+N)+PHI2P
PHI2N=XETA2*NN*GRHS(3*NT+N)+PHI2N
170 CONTINUE
PHI1D=PHI1DP+PHI1DN+GRHS(3*NT+1)/Z11
PHI2D=PHI2DP+PHI2DN+GRHS(3*NT+2)/Z22
PHI1=PHI1P+PHI1N+GRHS(3*NT+1)*CDLOG(XETA1)
PHI2=PHI2P+PHI2N+GRHS(3*NT+2)*CDLOG(XETA2)
SGMAX=2.*(R1*R1*PHI1D+R2*R2*PHI2D)
SGMAY=2.*(PHI1D+PHI2D)
SGMAXY=-2.*(R1*PHI1D+R2*PHI2D)
EPSX=AMAT(1,1)*SGMAX+AMAT(1,2)*SGMAY+AMAT(1,3)*SGMAXY
EPSY=AMAT(2,1)*SGMAX+AMAT(2,2)*SGMAY+AMAT(2,3)*SGMAXY
EPSXY=AMAT(3,1)*SGMAX+AMAT(3,2)*SGMAY+AMAT(3,3)*SGMAXY
U=2.*(P1*PHI1+P2*PHI2)
V=2.*(Q1*PHI1+Q2*PHI2)
PI=DARCO5(-1.00)
IF(XOUT(K).GT.0..AND.YOUT(K).GT.0.)
+TETAA=DATAN(YOUT(K)/XOUT(K))*180./PI
IF(XOUT(K).LT.0..AND.YOUT(K).GT.0.)
+TETAA=DATAN(YOUT(K)/XOUT(K))*180./PI+180.
IF(XOUT(K).LT.0..AND.YOUT(K).LT.0.)
+TETAA=DATAN(YOUT(K)/XOUT(K))*180./PI+180.
IF(YOUT(K).LT.0..AND.XOUT(K).GT.0.)
+TETAA=DATAN(YOUT(K)/XOUT(K))*180./PI+360.
C=DCOS(TETAA*PI/180.)
S=DSIN(TETAA*PI/180.)
SGMAR=C**2*SGMAX+S**2*SGMAY+2.*C*S*SGMAXY
SGMAT=S**2*SGMAX+C**2*SGMAY-2.*C*S*SGMAXY
SGMART=-C*S*SGMAX+C*S*SGMAY+(C**2-S**2)*SGMAXY
EPSR=C**2*EPSX+S**2*EPSY+C*S*EPSXY
EPST=S**2*EPSX+C**2*EPSY-C*S*EPSXY
EPSRT=2.*(-C*S*EPSX+C*S*EPSY+(C**2-S**2)*(EPSXY/2.))
UR=UXC+V*XS
IF(NOPT4.EQ.5) GO TO 3338
KTHT(K)=TETAA
REPX(K)=EPSX
REPY(K)=EPSY
REPLY(K)=EPSXY
ASX(K)=SGMAX
00023960
00023970
00023980
00023990
00024000
00024010
00024020
00024030
00024040
00024050
00024060
00024070
00024080
00024090
00024100
00024110
00024120
00024130
00024140
00024150
00024160
00024170
00024180
00024190
00024200
00024210
00024220
00024230
00024240
00024250
00024260
00024270
00024280
00024290
00024300
00024310
00024320
00024330
00024340
00024350
00024360
00024370
00024380
00024390
00024400
00024410
00024420
00024430
00024440
00024450
00024460
00024470
00024480
00024490
00024500
00024510
00024520
00024530
00024540
00024550

```

	ASXY(K)=SGMAXY	00024560
	FUR(X)=UR	00024570
	FTHT(K)=TETAA	00024580
	FSMR(K)=SGMAR	00024590
3338	CONTINUE	00024600
	IF(NOPT4.EQ.5.AND.K.GT.NOUT) GO TO 3339	00024610
	APSX(K)=SGMAX	00024620
	APSY(K)=SGMAX	00024630
	APSY(K)=SGMAX	00024640
3339	CONTINUE	00024650
	IF(NOPT4.EQ.1) GO TO 190	00024660
	IF(NOPT4.EQ.5.AND.K.LE.NOUT.OR.K.GT.(NOUT+NSTS)) GO TO 191	00024670
	IF(IC2.LE.NINC) STSV(IC2)=SGMAX	00024680
	IF(IC2.GT.NINC.AND.IC2.LE.(2*NINC)) STSV(IC2)=SGMAX	00024690
	IF(IC2.GT.(2*NINC).AND.IC2.LE.(3*NINC)) STSV(IC2)=SGMAX	00024700
	IF(IC2.GT.(3*NINC).AND.IC2.LE.(4*NINC)) STSV(IC2)=SGMAX	00024710
	IC2=IC2+1	00024720
	GO TO 190	00024730
191	CONTINUE	00024740
	IF(NIC.EQ.1) GO TO 192	00024750
	IF(NOPT4.EQ.5.AND.X.LT.(NRCF-7)) GO TO 190	00024760
	UVOUT(IC)=U	00024770
	UVOUT(IC+1)=V	00024780
	IC=IC+2	00024790
	GO TO 190	00024800
192	CONTINUE	00024810
	NC=NOUT+NSTS	00024820
	IF(K.GT.NC.AND.K.LE.(NC+NCOL)) SUMU1=SUMU1+U	00024830
	IF(K.GT.(NC+NCOL).AND.K.LE.(NC+2*NCOL)) SUMV1=SUMV1+V	00024840
	IF(K.GT.(NC+2*NCOL).AND.K.LE.(NC+3*NCOL)) SUMU2=SUMU2+U	00024850
	IF(K.GT.(NC+3*NCOL).AND.K.LE.(NC+4*NCOL)) SUMV2=SUMV2+V	00024860
	NNC=NC+4*NCOL	00024870
	IF(K.EQ.(NNC+1)) UVOUT(9)=U	00024880
	IF(K.EQ.(NNC+1)) UVOUT(10)=V	00024890
	IF(K.EQ.(NNC+2)) UVOUT(11)=U	00024900
	IF(K.EQ.(NNC+2)) UVOUT(12)=V	00024910
	IF(K.EQ.(NNC+3)) UVOUT(13)=U	00024920
	IF(K.EQ.(NNC+3)) UVOUT(14)=V	00024930
	IF(K.EQ.(NNC+4)) UVOUT(15)=U	00024940
	IF(K.EQ.(NNC+4)) UVOUT(16)=V	00024950
190	CONTINUE	00024960
	DISPLACEMENTS ARE AVERAGED OVER ELEMENT SIDES FOR	00024970
	CERTAIN LOAD CASES	00024980
	IF(NIC.NE.1) RETURN	00024990
	SUMU1=SUMU1/FLOAT(NCOL)	00025000
	SUMV1=SUMV1/FLOAT(NCOL)	00025010
	SUMU2=SUMU2/FLOAT(NCOL)	00025020
	SUMV2=SUMV2/FLOAT(NCOL)	00025030
	UVOUT(1)=SUMU2	00025040
	UVOUT(2)=SUMV2	00025050
	UVOUT(3)=SUMU2	00025060
	UVOUT(4)=SUMV1	00025070
	UVOUT(5)=SUMU1	00025080
	UVOUT(6)=SUMV1	00025090
	UVOUT(7)=SUMU1	00025100
	UVOUT(8)=SUMV2	00025110
	RETURN	00025120
	END	00025130
		00025140
		00025150

00000000

000

00000000

SUBROUTINE FBOLT(ANGK,H,PSH,K)

FBOLT CALCULATES THE INDIVIDUAL PLY FOUNDATION
MODULI AND THE INDIVIDUAL PLY LOADS

IMPLICIT REAL*8(A-H,O-Z)
DIMENSION ATETAA(400),ANG(5,2),ASIGR(400),ASIGRT(400),H(2)
DIMENSION ASIG1(400),ASIG2(400),ASIG6(400),UR(400),ANGK(5,2)
DIMENSION FSMR(400),PLXPT(100)
DIMENSION IPLY(100,2),NPLY(2),NUMPLY(2)
DIMENSION FKI(100),PLX(100)
DIMENSION E11(2),E22(2),ESS(2),PMU12(2),PMU21(2)
COMMON/STRESS/ASIGR,ASIGRT,ASIG1,ASIG2,ASIG6
COMMON/ELP/AX,BX,HOUT
COMMON/FB1/BSTR,YSTR
COMMON/LYP/NPLY,NUMPLY,ANG,IPLY
COMMON/FB2/UR,ATETAA,FSMR
COMMON/MOD/E11,E22,ESS,PMU12,PMU21
COMMON/FCT/PLXPT
RAD=DARCOS(-0.1D1)/180.
THKTOT=NPLY(K)*H(K)
NN=NUMPLY(K)

CALCULATE DELEFF

WOPK=0.
PLOADX=0.
IF(K.EQ.1) PLD=0.
DO 210 KK=1,NOUT
TH1=ATETAA(KK+1)-ATETAA(KK)
TH2=(ATETAA(KK)+ATETAA(KK+1))/2.
THETA=TH2*RAD
C=DCOS(THETA)
S=DSIN(THETA)
R=DSQRT(1./((C**2/AX**2+S**2/BX**2))
FORCE=((FSMR(KK)+FSMR(KK+1))/2.)*R*TH1*RAD*THKTOT
WORK=WORK+FORCE*.5*((UR(KK)+UR(KK+1))/2.)
PLOADX=PLOADX+FORCE*C

210 CONTINUE
PLD=PLD+PLOADX
DELEF=WORK/PLOADX

COMPUTE PLY STRESSES FROM LAMINATE STRAINS

(SIGMA)R,0,R0 = (Q)*(EPS)R,0,R0

NN=NPLY(K)
DO 100 J=1,NN
LP=IPLY(J,K)
THETA=(ANG(LP,K)+PSH)*RAD
I1=1

00025160
00025170
00025180
00025190
00025200
00025210
00025220
00025230
00025240
00025250
00025260
00025270
00025280
00025290
00025300
00025310
00025320
00025330
00025340
00025350
00025360
00025370
00025380
00025390
00025400
00025410
00025420
00025430
00025440
00025450
00025460
00025470
00025480
00025490
00025500
00025510
00025520
00025530
00025540
00025550
00025560
00025570
00025580
00025590
00025600
00025610
00025620
00025630
00025640
00025650
00025660
00025670
00025680
00025690
00025700
00025710
00025720
00025730
00025740
00025750

00000000

LI2=NOUT
NCAS=1
CALL QMATX(RAD,THETA,K,LI1,LI2,NCAS)

INTEGRATE AROUND CIRCULAR BOUNDARY FOR
INDIVIDUAL PLY LOADS AND COMPUTE FOUNDATION
MODULI

```

NNH=LI2-1
PLOADX=0.
WK=0.
DO 70 I=LI1,NNH
  TH1=ATETAA(I+1)-ATETAA(I)
  TH2=(ATETAA(I)+ATETAA(I+1))/2.
  THETA=TH2*H*AD
  C=DCOS(THETA)
  S=DSIN(THETA)
  R=DSQRT(1./(C**2/AX**2+S**2/BX**2))
  FORCR=((ASIGR(I)+ASIGR(I+1))/2.)*R*TH1*RAD*H(K)
  FORCRT=((ASIORT(I)+ASIORT(I+1))/2.)*R*TH1*RAD*H(K)
  PLOADX=PLOADX+FORCR*C-FORCRT*S
70 CONTINUE
FKI(J)=DABS(PLOADX/(H(K)*DELEFF))
PLX(J+(K-1)*NPLY(1))=I*PLOADX
100 CONTINUE
NT=NPLY(K)
NN=NPLY(K)
DO 310 I=1,NT
  DO 310 II=1,NN
    IF(IPLY(II,K).EQ.I) A'IGK(I,K)=FKI(II)
    IF(IPLY(II,K).EQ.I) PLXPT(II)=PLX(II+(K-1)*NPLY(1))
310 CONTINUE
NP=NPLY(K)
DO 311 I=1,NP
  AA=ANG(I,K)+PSH
311 CONTINUE
PLXTOT=0.000
IF(K.EQ.1) BLOAD=0.
TH=H(K)*NPLY(K)
BLOAD=(BSTR*DARCOS(-1.000)*BX*TH)/2.+BLOAD
IF(K.EQ.1) GO TO 611
NN=NPLY(1)+NPLY(2)
DO 212 I=1,NN
  PLXTOT=PLXTOT+PLX(I)
212 CONTINUE
611 CONTINUE
RETURN
END

```

00000000

SUBROUTINE QMATX(RAD,THETA,K,LI1,LI2,NCAS)

QMATX PERFORMS BASIC STRESS AND STRAIN
TRANSFORMATIONS

IMPLICIT REAL*8(A-H,O-Z)
DIMENSION ASIGR(400),ASIORT(400),ASIG1(400),ASIG2(400),ASIG6(400)

00025760
00025770
00025780
00025790
00025800
00025810
00025820
00025830
00025840
00025850
00025860
00025870
00025880
00025890
00025900
00025910
00025920
00025930
00025940
00025950
00025960
00025970
00025980
00025990
00026000
00026010
00026020
00026030
00026040
00026050
00026060
00026070
00026080
00026090
00026100
00026110
00026120
00026130
00026140
00026150
00026160
00026170
00026180
00026190
00026200
00026210
00026220
00026230
00026240
00026250
00026260
00026270
00026280
00026290
00026300
00026310
00026320
00026330
00026340
00026350

```

DIMENSION ATETAA(400),AEP SX(400),AEP SY(400),AEP SXY(400) 00026360
DIMENSION E11(2),E22(2),ESS(2),PMU12(2),PMU21(2),SX(400),SXY(400) 00026370
DIMENSION AEP S1(400) 00026380
DIMENSION ASX(400),ASXY(400) 00026390
COMMON/XXY1/ASX,ASXY 00026400
COMMON/MOD/E11,E22,ESS,PMU12,PMU21 00026410
COMMON/STRESS2/AEP S1 00026420
COMMON/STRESS/ASIOR,ASIOR1,ASIOR2,ASIOR3,ASIOR4 00026430
COMMON/CMT/ATETAA,AEP SX,AEP SY,AEP SXY 00026440
COMMON/PSCS/SX,SXY 00026450
J=0 00026460
Q11=E11(K)/(1.0-PMU12(K)*PMU21(K)) 00026470
Q12=(PMU21(K)*E11(K))/(1.0-PMU12(K)*PMU21(K)) 00026480
Q22=E22(K)/(1.0-PMU12(K)*PMU21(K)) 00026490
Q66=ESS(K) 00026500
C=DCOS(THETA) 00026510
S=DSIN(THETA) 00026520
BQ11=Q11*(C**4)+(2.*(Q12+(2.*Q66))*C**2*(S**2))+Q22*(S**4) 00026530
BQ12=-(Q11+Q22-(4.*Q66))*C**2*(S**2)+(Q12*(S**4+C**4)) 00026540
BQ14=-(Q11-Q12-(2.*Q66))*C*(S**3)+((Q12-Q22+(2.*Q66))*C*(S**3)*C 00026550
*) 00026560
BQ22=(Q11*(S**4)+(2.*(Q12+(2.*Q66))*C**2*(S**2))+Q22*(C**4)) 00026570
BQ26=-(Q11-Q12-(2.*Q66))*C*(S**3)+((Q12-Q22+(2.*Q66))*C*(S**3)*C 00026580
BQ46=-(Q11+Q22-(2.*(Q12+Q66))*C**2*(S**2))+Q66*(C**4)+(S**4) 00026590
*) 00026600
DO 40 I=111,112 00026610
J=J+1 00026620
IF(NCAS.EQ.1) THETA=ATETAA(I)*RAD 00026630
C=DCOS(THETA) 00026640
S=DSIN(THETA) 00026650
SIGX=BQ11*AEP SX(I)+BQ12*AEP SY(I)+BQ14*AEP SXY(I) 00026660
SIGY=BQ12*AEP SX(I)+BQ22*AEP SY(I)+BQ26*AEP SXY(I) 00026670
SIGXY=BQ14*AEP SX(I)+BQ26*AEP SY(I)+BQ46*AEP SXY(I) 00026680
SX(J)=SIGX 00026690
SXY(J)=SIGY 00026700
ASIGR(I)=SIGX*C**2+SIGY*S**2+2.*SIGXY*S*C 00026710
ASIGRT(I)=-SIGX*S*C+SIGY*C*S+SIGXY*(C**2-S**2) 00026720
ASIG1(J)=SIGX*C**2+SIGY*S**2+2.*S*C*SIGXY 00026730
ASIG2(J)=SIGX*S**2+SIGY*C**2-2.*S*C*SIGXY 00026740
ASIG6(J)=-C*S*SIGX+SIGY*C*S+(C**2-S**2)*SIGXY 00026750
AEP S1(J)=AEP SX(I)*C**2+AEP SY(I)*S**2+AEP SXY(I)*S*C 00026760
40 CONTINUE 00026770
RETURN 00026780
END 00026790
00026800
00026810
00026820
00026830
00026840
00026850
00026860
00026870
00026880
00026890
00026900
00026910
00026920
00026930
00026940
00026950

SUBROUTINE CENTD(H,FASSS,FASBS,P)

IMPLICIT REAL*8(A-H,O-Z)
DIMENSION PLYK(100),BARK(100),BARU(100),F(100)
DIMENSION H(2),RF(2)
DIMENSION AII(100,100),A(2),B(2)
DIMENSION NPLY(2)
COMMON/PBB/PLYK,BARK,BARU
COMMON/RT/RF
COMMON/AFM/AII,F

```


	COMMON/LYP/NPLY	00026960
	NNN=NPLY(1)+NPLY(2)	00026970
C	SET UP THE CENTRAL DIFFERENCE EQUATIONS	00026980
C	DO 3 I=1,100	00026990
	DO 3 J=1,100	00027000
C	3 AII(I,J)=0.	00027010
		00027020
C	NECESSARY CONSTANTS ARE FORMED	00027030
		00027040
C	DO 7 I=1,...	00027050
	A(I)=H(I)*H2/FASSS	00027060
	7 B(I)=H(I)*H4/FASBS	00027070
	H12=H(1)/H(2)	00027080
	A1=H(1)*H2/FASSS	00027090
	A2=H(2)*H2/FASSS	00027100
	NP=NPLY(1)+NPLY(2)	00027110
C		00027120
		00027130
C	SHEAR AT TOP OF PANEL EQUALS ZERO	00027140
		00027150
C	AII(1,1)=1.	00027160
	AII(1,2)=-2.+A1*PLYK(2)	00027170
	AII(1,4)=2.+A1*PLYK(2)	00027180
	AII(1,5)=-1.	00027190
	F(1)=0.0	00027200
C	MOMENT CONDITION AT TOP	00027210
		00027220
	IF(RF(1).OE.1.D10) GO TO 50	00027230
	Z=1.	00027240
	R=RF(1)	00027250
	GO TO 60	00027260
50	Z=0.	00027270
	R=1.	00027280
60	AII(2,1)=R	00027290
	AII(2,2)=(Z*2.*H(1)*FASSS)*RM(-2.-A1*PLYK(2)+(H(1)*H2	00027300
	*FASSS)/FASBS)	00027310
	AII(2,3)=-Z*(4.*H(1)*FASSS+(2*H(1)*H2*PLYK(1)*H(1)))	00027320
	AII(2,4)=Z*2.*H(1)*FASSS+RM(2.+A1*PLYK(2)-(H(1)*H2	00027330
	*FASSS)/FASBS)	00027340
	AII(2,5)=-R	00027350
	F(2)=Z*2.*H(1)*H3*BARK(1)*BARU(1)	00027360
C	GOVERNING EQUATIONS FOR THE TOP PLATE	00027370
		00027380
	N2=NPLY(1)	00027390
	DO 55 J=1,N2	00027400
	I=J+2	00027410
	AII(I,J)=1.	00027420
	IF(J.EQ.1) GO TO 56	00027430
	AII(I,J+1)=-4.-A(1)*PLYK(J-1)	00027440
	GO TO 57	00027450
56	AII(I,J+1)=-4.-A(1)*PLYK(2)	00027460
57	AII(I,J+2)=6.+(2.*A(1)+B(1))*PLYK(J)	00027470
	IF(J.EQ.N2) GO TO 61	00027480
	AII(I,J+3)=-4.-A(1)*PLYK(J+1)	00027490
	GO TO 62	00027500
61	AII(I,J+3)=-4.-A(1)*PLYK(NPLY(1)-1)	00027510
		00027520
		00027530
		00027540
		00027550

62	ALL(I,J+4)=1.	00027560
	IF(J.EQ.N2) GO TO 58	00027570
	IF(J.EQ.N2) GO TO 63	00027580
	F(I)=A(1)*BARK(J-1)*BARU(J-1)	00027590
	M=(2.*A(1)+B(1))*BARK(J)*BARU(J)	00027600
	M+A(1)*BARK(J+1)*BARU(J+1)	00027610
	GO TO 59	00027620
58	F(I)=2.*A(1)*BARK(2)*BARU(2)	00027630
	M=(2.*A(1)+B(1))*BARK(1)*BARU(1)	00027640
	GO TO 59	00027650
63	F(I)=2.*A(1)*BARK(NPLY(1)-1)*BARU(NPLY(1)-1)	00027660
	M=(2.*A(1)+B(1))*BARK(J)*BARU(J)	00027670
59	CONTINUE	00027680
55	CONTINUE	00027690
C	INTERFACE SHEAR ON TOP PLATE = P	00027700
C	I=NPLY(1)+3	00027710
	J=NPLY(1)	00027720
	ALL(I,J)=1.	00027730
	ALL(I,J+1)=-((2.+A1*PLYK(NPLY(1)-1))	00027740
	ALL(I,J+3)=2.+A1*PLYK(NPLY(1)-1)	00027750
	ALL(I,J+4)=-1.	00027760
	F(I)=(-2.*H(1)*M3*(P))/FASBS	00027770
C	SLOPE CONTINUITY	00027780
C	I=NPLY(1)+4	00027790
	J=NPLY(1)	00027800
	ALL(I,J)=1.	00027810
	ALL(I,J+1)=-((2.+A1*PLYK(NPLY(1)-1)-H(1)*M2*FASBS/FASBS)	00027820
	ALL(I,J+3)=2.+A1*PLYK(NPLY(1)-1)-H(1)*M2*FASBS/FASBS	00027830
	ALL(I,J+4)=-1.	00027840
	ALL(I,J+5)=-H12*M3	00027850
	ALL(I,J+6)=H12*M3*(2.+A2*PLYK(NPLY(1)+2)-H(2)*M2*FASBS/FASBS)	00027860
	ALL(I,J+8)=-H12*M3*(2.+A2*PLYK(NPLY(1)+2)-H(2)*M2*FASBS/FASBS)	00027870
	ALL(I,J+9)=H12*M3	00027880
	F(I)=0.	00027890
C	MOMENT CONTINUITY	00027900
C	I=NPLY(1)+5	00027910
	J=NPLY(1)+1	00027920
	ALL(I,J)=1.	00027930
	ALL(I,J+1)=-((2.+A1*PLYK(NPLY(1)))	00027940
	ALL(I,J+2)=1.	00027950
	ALL(I,J+5)=-H12*M2	00027960
	ALL(I,J+6)=H12*M2*(2.+A2*PLYK(NPLY(1)+1))	00027970
	ALL(I,J+7)=-H12*M2	00027980
	F(I)=A1*(BARK(NPLY(1))*BARU(NPLY(1))-BARK(NPLY(1)+1)*	00027990
	BARU(NPLY(1)+1))	00028000
C	INTERFACE SHEAR ON BOTTOM PLATE	00028010
C	I=NPLY(1)+6	00028020
	J=NPLY(1)+5	00028030
	ALL(I,J)=-1.	00028040
	ALL(I,J+1)=(2.+A2*PLYK(NPLY(1)+2))	00028050
	ALL(I,J+3)=-((2.+A2*PLYK(NPLY(1)+2))	00028060
	ALL(I,J+4)=1.	00028070
		00028080
		00028090
		00028100
		00028110
		00028120
		00028130
		00028140
		00028150

	F(I)=2.*H(2)*NPLY(1)*333	00028140
CC	GOVERNING EQUATIONS FOR THE BOTTOM PLATE	00028170
	N1=NPLY(1)+7	00028180
	N2=NPLY(1)+NPLY(2)+6	00028190
	DO 70 I=N1,12	00028200
	J=I-2	00028210
	ALL(I,J)=1.	00028220
	IF(I.EQ.N1) GO TO 71	00028230
	ALL(I,J-1)=-4.-1(2)*PLYK(J-5)	00028240
	GO TO 72	00028250
71	ALL(I,J+1)=-4.-1(2)*PLYK(NPLY(1)+2)	00028260
72	ALL(I,J+2)=6.+(2.*A(2)+B(2))*PLYK(J-4)	00028270
	IF(I.EQ.N2) GO TO 75	00028280
	ALL(I,J-3)=-4.-1(2)*PLYK(J-3)	00028290
	GO TO 73	00028300
75	ALL(I,J+3)=-4.-1(2)*PLYK(J-5)	00028310
73	ALL(I,J+4)=1.	00028320
	IF(I.EQ.N1) GO TO 73	00028330
	IF(I.EQ.N2) GO TO 77	00028340
	F(I)=A(2)*BARK(J-5)*BARU(J-5)	00028350
	X=(2.*A(2)+B(2))*BARK(J-4)*BARU(J-4)	00028360
	+A(2)*BARK(J-3)*BARU(J-3)	00028370
	GO TO 74	00028380
73	F(I)=2.*A(2)*BARK(NPLY(1)+2)*BARU(NPLY(1)+2)	00028390
	X=(2.*A(2)+B(2))*BARK(J-4)*BARU(J-4)	00028400
	GO TO 74	00028410
77	F(I)=2.*A(2)*BARK(J-5)*BARU(J-5)	00028420
	X=(2.*A(2)+B(2))*BARK(J-4)*BARU(J-4)	00028430
74	CONTINUE	00028440
70	CONTINUE	00028450
CC	SHEAR ON BOTTOM PLATE EQUALS ZERO	00028460
	NP=NPLY(1)+NPLY(2)	00028470
	I=NP+7	00028480
	J=NP+4	00028490
	ALL(I,J)=-1.	00028500
	ALL(I,J+1)=(2.+A2*PLYK(NP-1))	00028510
	ALL(I,J+3)=-2.+A2*PLYK(NP-1)	00028520
	ALL(I,J+4)=1.	00028530
	F(I)=0.	00028540
CC	MOMENT BOUNDARY CONDITION ON BOTTOM PLATE	00028550
	I=NP+8	00028560
	IF(RF(2).GE.1.D10) GO TO 85	00028570
	Z=1.	00028580
	R=RF(2)	00028590
	GO TO 95	00028600
85	Z=0.	00028610
	R=1.	00028620
95	ALL(I,J)=-R	00028630
	ALL(I,J+1)=Z*(2.*H(2)*FAS55)+R*(2.+A2*PLYK(NP-1))	00028640
	X=H(2)*M2*FAS55/FAS85	00028650
	ALL(I,J+2)=-Z*(4.*H(2)*FAS55+2.*H(2)*M3*PLYK(NP))	00028660
	ALL(I,J+3)=Z*2.*H(2)*FAS55+R*(-2.-A2*PLYK(NP-1))	00028670
	X=H(2)*M2*FAS55/FAS85	00028680
	ALL(I,J+4)=R	00028690
		00028700
		00028710
		00028720
		00028730
		00028740
		00028750

	F(I)=Z*(2.*H(2)*X3+JARK(NP)*BARU(NP))	00028760
	RETURN	00023770
	END	00023780
		00028790
	SUBROUTINE SOLVE(H,P,U1,U2)	00028800
		00028810
		00028820
		00028830
		00028840
		00028850
	IMPLICIT REAL*8(A-H,O-Z)	00023860
	DIMENSION A(100,100),B(100),NPLY(2),U(100),F(100)	00023370
	DIMENSION SX(100),PLYX(100),H(2)	00023880
	DIMENSION JARK(100),BARU(100)	00028890
	COMMON/PLP/NPLY	00023900
	COMMON/AFM/A,F	00023910
	COMMON/P33/PLYX,JARK,BARU	00023920
		00023930
	SOLUTION OF THE SYSTEM: PA(U)=(3)	00023940
	NP=NPLY(1)+NPLY(2)+8	00023950
	DO 444 I=1,NP	00023960
444	B(I)=F(I)	00028970
		00028980
		00028990
	APPLYING GUASSIAN ELIMINATION TO THE	00029000
	MATRIX OF COEFFICIENTS	00029010
		00029020
	DO 2001 I=1,NP	00029030
	IR=I	00029040
2042	IF(A(IR,I).NE.0.) GO TO 2041	00029050
	IR=IR+1	00029060
	IF(IR.GT.NP) GO TO 2001	00029070
	GO TO 2042	00029080
2041	NN=IR+1	00029090
	DO 2002 L=NN,NP	00029100
	IF(DABS(A(L,I)).GT.1.D-30) GO TO 2009	00029110
	A(L,I)=0.	00029120
	GO TO 2002	00029130
2009	CF=-A(IR,I)/A(L,I)	00029140
	DO 2003 J=I,NP	00029150
	A(L,J)=A(L,J)+CF*A(IR,J)	00029160
	IF(DABS(A(L,J)).LT.1.D-30) A(L,J)=0.0	00029170
2003	CONTINUE	00029180
	B(L)=B(L)+CF*B(I)	00029190
2002	CONTINUE	00029200
2001	CONTINUE	00029210
		00029220
	BACK SUBSTITUTION	00029230
		00029240
	DO 2011 I=1,NP	00029250
	L=NP+1-I	00029260
	SUM=0.	00029270
	IF(A(L,L).EQ.0.) GO TO 2112	00029280
	N=L+1	00029290
	IF(N.GT.NP) GO TO 2013	00029300
	DO 2013 J=N,NP	00029310
	SUM=SUM-A(L,J)*SX(J)	00029320
2013	CONTINUE	00029330
	SX(L)=(B(L)+SUM)/A(L,L)	00029340
	GO TO 2011	00029350

2112	CONTINUE	00029360
	SX(L)=0.	00029370
2011	CONTINUE	00029380
	PT=P	00029390
	N1=NPLY(1)+2	00029400
	N2=NPLY(1)+7	00029410
	NN=NPLY(1)+NPLY(2)+6	00029420
	DO 1444 I=3,N1	00029430
	J=I-2	00029440
	U(J)=SX(I)	00029450
1444	CONTINUE	00029460
	DO 1555 I=N2,NN	00029470
	J=I-6	00029480
	U(J)=SX(I)	00029490
1555	CONTINUE	00029500
	NP=NPLY(1)+NPLY(2)	00029510
		00029520
	COMPUTE AVERAGE RELATIVE DISPLACEMENTS	00029530
	IN TOP AND BOTTOM PLATES	00029540
		00029550
	U1=DABS(U(1)-U(NPLY(1)))/2.	00029560
	U2=DABS(U(NPLY(1)+1)-U(NPLY(1)+NPLY(2)))/2.	00029570
	RETURN	00029580
	END	00029590
		00029600
		00029610
		00029620
	SUBROUTINE FCRT(APP,NEL1,NEL2,NDAM,IN,LTNCM,NAVD)	00029630
	IMPLICIT REAL*8(A-H,O-Z)	00029640
	DIMENSION ELSTFF(50,10,10),ELCTSS(50,50,10),U(200)	00029650
	DIMENSION OSSX(20),OSSW(20)	00029660
	DIMENSION NELDIS(50,5,2)	00029670
	DIMENSION PSMX(50,4),AVES(50,3),STRSS(50),DLT(10)	00029680
	DIMENSION ELFAIL(50,3),NELTYP(50)	00029690
	DIMENSION NELCON(50,6),NELCHA(50,6),NPLY(2)	00029700
	DIMENSION ELWDTH(50),ELTHK(50),ELLOAD(50,2)	00029710
	DIMENSION NELPLS(2,50),LYPN(50)	00029720
	COMMON/ELS/ELSTFF,ELSTSS	00029730
	COMMON/OSXW/OSSX,OSSW	00029740
	COMMON/NPLS/NELPLS,LYPN	00029750
	COMMON/NCH/NELCON,NELCHA,NELDIS	00029760
	COMMON/FCC/ELWDTH,ELTHK,ELLOAD	00029770
	COMMON/SMX/PSMX	00029780
	COMMON/LAMF/ELFAIL	00029790
	COMMON/DISP/U	00029800
	COMMON/NTP/NELTYP	00029810
	COMMON/LYP/NPLY	00029820
		00029830
	DETERMINE ELEMENT FAILURE LOADS IN NET SECTION,	00029840
	SHEAROUT AND BEARING, AND LOCATE THE CRITICAL	00029850
	FASTENER LOCATION. JOINT STRENGTH IS DETERMINED	00029860
	FROM LOWEST ELEMENT FAILURE LOAD	00029870
		00029880
	NELTOT=NEL1+NEL2	00029890
	NS=NAVD	00029900
	NSTS=4*NAVD	00029910
	DO 10 I=1,NELTOT	00029920
	NRNK=10	00029930
	IF(NELTYP(I).EQ.3) NRNK=8	00029940
	KJ=I	00029950

IF(KJ.GT.NEL) KJ=KJ-NEL	C0029960
IF(NELTYP(I).EQ.1) GO TO 10	00029970
IC=0	00029980
DO 20 J=1,5	00029990
IC=IC+1	00030000
DLT(IC)=U(NELDIS(I,J,1))	00030010
IC=IC+1	00030020
DLT(IC)=U(NELDIS(I,J,2))	00030030
20 CONTINUE	00030040
DO 30 K=1,NSTS	00030050
SUM=0.000	00030060
DO 40 K2=1,NRNK	00030070
SUM=SUM+ELSTS3(I,K,K2)*DLT(K2)	00030080
40 CONTINUE	00030090
30 STRSS(K)=SUM	00030100
SUM1=0.000	00030110
SUM2=0.000	00030120
SUM3=0.000	00030130
SUM4=0.000	00030140
DO 50 J=1,NS	00030150
SUM1=SUM1+STRSS(J)	00030160
SUM2=SUM2+STRSS(J+NS)	00030170
SUM3=SUM3+STRSS(J+2*NS)	00030180
50 CONTINUE	00030190
NR=2*NS	00030200
DO 51 II=1,NS	00030210
51 SUM4=SUM4+STRSS(II+NS)*MOSSW(II)	00030220
AVES(I,1)=SUM1/NS	00030230
AVES(I,2)=SUM2/NS	00030240
AVES(I,3)=SUM3/NS	00030250
IF(I.LE.NEL1) THK=ELTHK(I)*NELPLS(1,LYPN(I))	00030260
IF(I.GT.NEL1) THK=ELTHK(I)*NELPLS(2,LYPN(I))	00030270
ELD=(SUM4/2.00)*ELHDTH(I)*THK	00030280
PRATIO=DABS((ELLOAD(I,1)+ELLOAD(I,2))/ELD)	00030290
C	00030300
C	00030310
C	00030320
SCALE AVERAGE STRESSES	00030330
AVES(I,1)=AVES(I,1)*PRATIO	00030340
AVES(I,2)=AVES(I,2)*PRATIO	00030350
AVES(I,3)=AVES(I,3)*PRATIO	00030360
10 CONTINUE	00030370
C	00030380
C	00030390
C	00030400
COMPUTE JOINT FAILURE LOADS BASED ON	00030410
ELEMENT LOADS	00030420
DO 100 I=1,NELTOT	00030430
IF(NELTYP(I).EQ.1) GO TO 100	00030440
DO 110 J=1,3	00030450
N=J+1	00030460
IF(J.EQ.1.AND.LTNCM.EQ.1) N=1	00030470
IF(J.EQ.1.AND.LTNCM.EQ.2) N=2	00030480
ELFAIL(I,J)=DABS(APP*PSMX(I,N)/AVES(I,J))	00030490
110 CONTINUE	00030500
100 CONTINUE	00030510
C	00030520
C	00030530
C	00030540
SEARCH FOR LOWEST JOINT FAILURE LOAD	00030550
INNS=0	
FNS=1.0D10	
INCO=0	

FSO=1.0D10	00030560
INBR=0	00030570
FBR=1.0D10	00030580
WRITE(6,356)	00030590
356 FORMAT(/, 'JOINT LOAD LEVELS CORRESPONDING TO NET ',/,	00030600
* SECTION (NS), SHEAR-OUT (SO) AND BEARING (BR)',/,	00030610
* FAILURES AT EVERY LOADED AND UNLOADED HOLE ',/,	00030620
* ELEMENT ARE PREDICTED AS FOLLOWS',/,	00030630
*3X, 'ELEMENT NS SO BR',/)	00030640
DO 120 I=1,NELTCT	00030650
IF(NELTYP(I).EQ.1) GO TO 120	00030660
IF(FNS.GT.DABS(ELFAIL(I,1))) INNS=I	00030670
IF(FNS.GT.DABS(ELFAIL(I,1))) FNS=DABS(ELFAIL(I,1))	00030680
IF(FSO.GT.DABS(ELFAIL(I,2))) INSO=I	00030690
IF(FSO.GT.DABS(ELFAIL(I,2))) FSO=DABS(ELFAIL(I,2))	00030700
IF(FBR.GT.DABS(ELFAIL(I,3))) INBR=I	00030710
IF(FBR.GT.DABS(ELFAIL(I,3))) FBR=DABS(ELFAIL(I,3))	00030720
WRITE(6,222) NELCON(I,1),ELFAIL(I,1),ELFAIL(I,2),ELFAIL(I,3)	00030730
222 FORMAT(2X,I8,2X,3(D9.3,2X))	00030740
120 CONTINUE	00030750
IF(FNS.GT.FSO.OR.FNS.GT.FBR) GO TO 130	00030760
NDAM=1	00030770
IN=INNS	00030780
GO TO 200	00030790
130 IF(FSO.GT.FNS.OR.FSO.GT.FBR) GO TO 140	00030800
NDAM=2	00030810
IN=INSO	00030820
GO TO 200	00030830
140 IF(FBR.GT.FNS.OR.FBR.GT.FSO) GO TO 200	00030840
NDAM=3	00030850
IN=INBR	00030860
200 CONTINUE	00030870
RETURN	00030880
END	00030890
C	00030900
C	00030910
C	00030920
SUBROUTINE LINV2F (A,N,IA,AINV,IDOT,WKAREA,IER)	00030930
C	00030940
DOUBLE PRECISION A(IA,N),AINV(IA,N),WKAREA(1),ZERO,ONE	00030950
DATA ONE/1.0D0/,ZERO/0.0D0/	00030960
C	00030970
FIRST EXECUTABLE STATEMENT	00030980
C	00030990
IER=0	00031000
C	00031010
SET AINV TO THE N X N	00031020
C	00031030
IDENTITY MATRIX	00031040
DO 10 I = 1,N	00031050
DO 5 J = 1,N	00031060
AINV(I,J) = ZERO	00031070
5 CONTINUE	00031080
AINV(I,I) = ONE	00031090
10 CONTINUE	00031100
C	00031110
COMPUTE THE INVERSE OF A	00031120
CALL LEQT2F (A,N,N,IA,AINV,IDOT,WKAREA,IER)	00031130
IF (IER.EQ.0) GO TO 9005	00031140
9000 CONTINUE	00031150
CALL UERTST (IER,6HLINV2F)	
9005 RETURN	
END	
C	

C	SUBROUTINE LEQT2F (A,M,N,IA,B,IDOT,WKAREA,IER)	00031160
C	DIMENSION A(IA,1),B(IA,1),WKAREA(1)	00031170
C	DOUBLE PRECISION A,B,WKAREA,D1,D2,WA	00031180
C	FIRST EXECUTABLE STATEMENT	00031190
C	INITIALIZE IER	00031200
	IER=0	00031210
	JER=0	00031220
	J = NXN+1	00031230
	K = J+N	00031240
	MM = K+N	00031250
	KK = 0	00031260
	MM1 = MM-1	00031270
	JJ=1	00031280
	DO 5 L=1,N	00031290
	DO 5 I=1,N	00031300
	WKAREA(JJ)=A(I,L)	00031310
	JJ=JJ+1	00031320
5	CONTINUE	00031330
C	DECOMPOSE A	00031340
	CALL LUDATN (WKAREA,N,N,A,IA,IDOT,D1,D2,WKAREA(J),WKAREA(K),	00031350
	WA,IER)	00031360
	IF (IER.GT.128) GO TO 25	00031370
	IF (IDOT.EQ. 0 .OR. IER .NE. 0) KK = 1	00031380
	DO 15 I = 1,M	00031390
C	PERFORMS THE ELIMINATION PART OF	00031400
C	AX = B	00031410
C	CALL LUEL MN (A,IA,N,B(1,I),WKAREA(J),WKAREA(MM))	00031420
C	REFINEMENT OF SOLUTION TO AX = B	00031430
	IF (KK .NE. 0)	00031440
	CALL LUREFN (WKAREA,N,N,A,IA,B(1,I),IDOT,WKAREA(J),WKAREA(MM),	00031450
	WKAREA(K),WKAREA(K),JER)	00031460
	DO 10 II=1,N	00031470
	B(II,I) = WKAREA(MM1+II)	00031480
10	CONTINUE	00031490
	IF (JER.NE.0) GO TO 20	00031500
15	CONTINUE	00031510
	GO TO 25	00031520
20	IER = 131	00031530
25	JJ=1	00031540
	DO 30 J = 1,M	00031550
	DO 30 I = 1,M	00031560
	A(I,J)=WKAREA(JJ)	00031570
	JJ=JJ+1	00031580
30	CONTINUE	00031590
	IF (IER.EQ. 0) GO TO 9005	00031600
9000	CONTINUE	00031610
	CALL UERTST (IER,6HLEQT2F)	00031620
9005	RETURN	00031630
	END	00031640
C		00031650
C		00031660
C		00031670
C		00031680
C		00031690
C	SUBROUTINE LUDATF (A,LU,N,IA,IDOT,D1,D2,IPVT,EQUIL,WA,IER)	00031700
	DIMENSION A(IA,1),IU(IA,1),IPVT(1),EQUIL(1)	00031710
	DOUBLE PRECISION A,LU,D1,D2,EQUIL,WA,ZERO,ONE,FOUR,SIXTN,SIXTH,	00031720
	RN,WREL,BIGA,BIG,P,SUM,AI,WI,T,TEST,Q	00031730
	DATA ZERO,ONE,FOUR,SIXTN,SIXTH/0.DO,1.DO,4.DO,	00031740
		00031750

		16.D0,.0625D0/	00031760
C		FIRST EXECUTABLE STATEMENT	00031770
C		INITIALIZATION	00031780
	IER = 0		00031790
	RN = N		00031800
	WREL = ZERO		00031810
	D1 = ONE		00031820
	D2 = ZERO		00031830
	BIOA = ZERO		00031840
	DO 10 I=1,N		00031850
	BIO = ZERO		00031860
	DO 5 J=1,N		00031870
	P = A(I,J)		00031880
	L(I,J) = P		00031890
	P = DABS(P)		00031900
	I: (P .GT. BIO) BIO = P		00031910
5	CONTINUE		00031920
	IF (BIO .GT. BIOA) BIOA = BIO		00031930
	IF (BIO .EQ. ZERO) GO TO 110		00031940
	EQU(I) = ONE/BIO		00031950
10	CONTINUE		00031960
	DO 105 J=1,N		00031970
	JMI = J-1		00031980
	IF (JMI .LT. 1) GO TO 40		00031990
C		COMPUTE U(I,J), I=1,...,J-1	00032000
	DO 35 I=1,JMI		00032010
	SUM = LU(I,J)		00032020
	IMI = I-1		00032030
	IF (IDOT .EQ. 0) GO TO 25		00032040
C		WITH ACCURACY TEST	00032050
	AI = DABS(SUM)		00032060
	WI = ZERO		00032070
	IF (IMI .LT. 1) GO TO 20		00032080
	DO 15 K=1,IMI		00032090
	T = LU(I,K)*LU(K,J)		00032100
	SUM = SUM+T		00032110
	WI = WI+DABS(T)		00032120
15	CONTINUE		00032130
	LU(I,J) = SUM		00032140
20	WI = WI+DABS(SUM)		00032150
	IF (AI .EQ. ZERO) AI = BIOA		00032160
	TEST = WI/AI		00032170
	IF (TEST .GT. WREL) WREL = TEST		00032180
	GO TO 35		00032190
C		WITHOUT ACCURACY	00032200
25	IF (IMI .LT. 1) GO TO 35		00032210
	DO 30 K=1,IMI		00032220
	SUM = SUM-LU(I,K)*LU(K,J)		00032230
30	CONTINUE		00032240
	LU(I,J) = SUM		00032250
35	CONTINUE		00032260
40	P = ZERO		00032270
C		COMPUTE U(J,J) AND L(I,J), I=J+1,...	00032280
	DO 70 I=J,N		00032290
	SUM = LU(I,J)		00032300
	IF (IDOT .EQ. 0) GO TO 55		00032310
C		WITH ACCURACY TEST	00032320
	AI = DABS(SUM)		00032330
	WI = ZERO		00032340
	IF (JMI .LT. 1) GO TO 50		00032350

	DO 65 K=1,JM1	00032360
	T = LU(I,K)*LU(K,J)	00032370
	SUM = SUM-T	00032380
	WI = WI+DABS(T)	00032390
45	CONTINUE	00032400
	LU(I,J) = SUM	00032410
50	WI = WI+DABS(SUM)	00032420
	IF (AI .EQ. ZERO) AI = BIOA	00032430
	TEST = WI/AI	00032440
	IF (TEST .GT. WREL) WREL = TEST	00032450
	GO TO 65	00032460
C	WITHOUT ACCURACY TEST	00032470
55	IF (JM .LT. 1) GO TO 65	00032480
	DO 60 K=1,JM1	00032490
	SUM = SUM-LU(I,K)*LU(K,J)	00032500
60	CONTINUE	00032510
	LU(I,J) = SUM	00032520
65	Q = EQUIL(I)*DABS(SUM)	00032530
	IF (P .GE. Q) GO TO 70	00032540
	P = Q	00032550
	IMAX = I	00032560
70	CONTINUE	00032570
C	TEST FOR ALGORITHMIC SINGULARITY	00032580
	IF (RH+P .EQ. RH) GO TO 110	00032590
C	IF (J .EQ. IMAX) GO TO 80	00032600
	INTERCHANGE ROWS J AND IMAX	00032610
	D1 = -D1	00032620
	DO 75 K=1,N	00032630
	P = LU(IMAX,K)	00032640
	LU(IMAX,K) = LU(J,K)	00032650
	LU(J,K) = P	00032660
75	CONTINUE	00032670
	EQUIL(IMAX) = EQUIL(J)	00032680
80	IPVT(J) = IMAX	00032690
	D1 = D1*LU(J,J)	00032700
85	IF (DABS(D1) .LE. ONE) GO TO 90	00032710
	D1 = D1*SIXTH	00032720
	D2 = D2+FOUR	00032730
	GO TO 85	00032740
90	IF (DABS(D1) .GE. SIXTH) GO TO 95	00032750
	D1 = D1*SIXTH	00032760
	D2 = D2-FOUR	00032770
	GO TO 90	00032780
95	CONTINUE	00032790
	JPI = J+1	00032800
	IF (JPI .GT. N) GO TO 105	00032810
C	DIVIDE BY PIVOT ELEMENT U(J,J)	00032820
	P = LU(J,J)	00032830
	DO 100 I=JPI,N	00032840
	LU(I,J) = LU(I,J)/P	00032850
100	CONTINUE	00032860
105	CONTINUE	00032870
C	PERFORM ACCURACY TEST	00032880
	IF (IDGT .EQ. 0) GO TO 9005	00032890
	P = 3*N+3	00032900
	WA = P*WREL	00032910
	IF (WA+10.D0**(-IDGT) .NE. WA) GO TO 9005	00032920
	IER = 34	00032930
	GO TO 9000	00032940
C	ALGORITHMIC SINGULARITY	00032950

110	IER = 129	00032960
	D1 = ZERO	00032970
	D2 = ZERO	00032980
9000	CONTINUE	00032990
	PRINT ERRJR	00033000
	CALL UERTST(IER,6HLUDATF)	00033010
9005	RETURN	00033020
	END	00033030
C		00033040
C		00033050
	SUBROUTINE LUELMN (A,IA,N,B,APVT,X)	00033060
C		00033070
	DIMENSION A(IA,1),B(1),APVT(1),X(1)	00033080
	DOUBLE PRECISION A,B,X,SUM,APVT	00033090
C		00033100
	FIRST EXECUTABLE STATEMENT	00033110
	SOLVE LY = B FOR Y	00033120
	DO 5 I=1,N	00033130
5	X(I) = B(I)	00033140
	IW = 0	00033150
	DO 20 I=1,N	00033160
	IP = APVT(I)	00033170
	SUM = X(IP)	00033180
	X(IP) = X(I)	00033190
	IF (IW .EQ. 0) GO TO 15	00033200
	IM1 = I-1	00033210
	DO 10 J=IW,IM1	00033220
	SUM = SUM-A(I,J)*X(J)	00033230
10	CONTINUE	00033240
	GO TO 20	00033250
15	IF (SUM .NE. 0.DO) IW = I	00033260
20	X(I) = SUM	00033270
C		00033280
	SOLVE UX = Y FOR X	00033290
	DO 30 IB=1,N	00033300
	I = N+1-IB	00033310
	IP1 = I+1	00033320
	SUM = X(I)	00033330
	IF (IP1 .GT. N) GO TO 30	00033340
	DO 25 J=IP1,N	00033350
	SUM = SUM-A(I,J)*X(J)	00033360
25	CONTINUE	00033370
30	X(I) = SUM/A(I,I)	00033380
	RETURN	00033390
	END	00033400
C		00033410
C		00033420
	SUBROUTINE LUREFN (A,IA,N,UL,IUL,B,IDGT,APVT,X,RES,DX,IER)	00033430
C		00033440
	DIMENSION A(IA,1),UL(IUL,1),B(1),X(1),RES(1),DX(1)	00033450
	DIMENSION APVT(1)	00033460
	DIMENSION ACCXT(2)	00033470
	DOUBLE PRECISION A,ACCXT,B,UL,X,RES,DX,ZERO,XNORM,DXNORM,APVT	00033480
	DATA ITMAX/75/,ZERO/0.DO/	00033490
C		00033500
	FIRST EXECUTABLE STATEMENT	00033510
	IER=0	00033520
	XNORM = ZERO	00033530
	DO 10 I=1,N	00033540
	XNORM = DMAX1(XNORM,DABS(X(I)))	00033550
10	CONTINUE	
	IF (XNORM .NE. ZERO) GO TO 20	
	IDGT = 50	

	GO TO 9005	00033560
20	DO 45 ITER=1,ITMAX	00033570
	DO 30 I=1,N	00033580
	ACCXT(1) = 0.000	00033590
	ACCXT(2) = 0.000	00033600
	CALL VXADD(B(I),ACCXT)	00033610
	DO 25 J=1,N	00033620
	CALL VXMUL(-A(I,J),X(J),ACCXT)	00033630
25	CONTINUE	00033640
	CALL VXSTO(ACCXT,RES(I))	00033650
30	CONTINUE	00033660
	CALL LUELMN (UL,IUL,N,RES,APVT,DX)	00033670
	DXNORM = ZERO	00033680
	XNORM = ZERO	00033690
	DO 35 I=1,N	00033700
	X(I) = X(I) + DX(I)	00033710
	DXNORM = DMAX1(DXNORM,DABS(DX(I)))	00033720
	XNORM = DMAX1(XNORM,DABS(X(I)))	00033730
35	CONTINUE	00033740
	IF (ITER .NE. 1) GO TO 40	00033750
	IDOT = 50	00033760
	IF (DXNORM .NE. ZERO) IDOT = -DLOG10(DXNORM/XNORM)	00033770
40	IF (XNORM+DXNORM .EQ. XNORM) GO TO 9005	00033780
45	CONTINUE	00033790
C		00033800
	IER = 129	00033810
9000	CONTINUE	00033820
	CALL UERTST(IER,6HLUREFN)	00033830
9005	RETURN	00033840
	END	00033850
C		00033860
C		00033870
C		00033880
	SUBROUTINE UERTST (IER,NAME)	00033890
C		00033900
	INTEGER IER	00033910
	INTEGER NAME(1)	00033920
C		00033930
	SPECIFICATIONS FOR LOCAL VARIABLES	00033940
	INTEGER I,IEQ,IEQDF,IOUNIT,LEVEL,LEVOLD,NAMEQ(6),	00033950
	NAMSET(6),NAMUPK(6),NIN,NMTB	00033960
	DATA NAMSET/1HU,1HE,1HR,1HS,1HE,1HT/	00033970
	DATA NAMEQ/6*1H /	00033980
	DATA LEVEL/4/,IEQDF/0/,IEQ/1H*/	00033990
C		00034000
C		00034010
	CALL USPDK (NAME,6,NAMUPK,NMTB)	00034020
C		00034030
	CALL UGETIO(1,NIN,IOUNIT)	00034040
C		00034050
	CHECK IER	00034060
	IF (IER.GT.999) GO TO 25	00034070
	IF (IER.LT.-32) GO TO 35	00034080
	IF (IER.LE.128) GO TO 5	00034090
	IF (LEVEL.LT.1) GO TO 30	00034100
C		00034110
	PRINT TERMINAL MESSAGE	00034120
	IF (IEQDF.EQ.1) WRITE(IOUNIT,35) IER,NAMEQ,IEQ,NAMUPK	00034130
	IF (IEQDF.EQ.0) WRITE(IOUNIT,35) IER,NAMUPK	00034140
	GO TO 30	00034150
5	IF (IER.LE.64) GO TO 10	00034160
	IF (LEVEL.LT.2) GO TO 30	00034170
C		00034180
	PRINT WARNING WITH FIX MESSAGE	00034190

	IF (IEQDF.EQ.1) WRITE(IUNIT,40) IER,NAMEQ,IEQ,NAMUPK	00034160
	IF (IEQDF.EQ.1) WRITE(IUNIT,40) IER,NAMUPK	00034170
	GO TO 30	00034180
C	10 IF (IER.LE.32) GO TO 15	00034190
	PRINT WARNING MESSAGE	00034200
	IF (LEVEL.LT.3) GO TO 30	00034210
	IF (IEQDF.EQ.1) WRITE(IUNIT,45) IER,NAMEQ,IEQ,NAMUPK	00034220
	IF (IEQDF.EQ.0) WRITE(IUNIT,45) IER,NAMUPK	00034230
	GO TO 30	00034240
	15 CONTINUE	00034250
C	CHECK FOR UERSET CALL	00034260
	DO 20 I=1,6	00034270
	IF (NAMUPK(I).NE.NAMSET(I)) GO TO 25	00034280
	20 CONTINUE	00034290
	LEVOLD = LEVEL	00034300
	LEVEL = IER	00034310
	IER = LEVOLD	00034320
	IF (LEVEL.LT.0) LEVEL = 4	00034330
	IF (LEVEL.GT.4) LEVEL = 4	00034340
	GO TO 30	00034350
	25 CONTINUE	00034360
	IF (LEVEL.LT.4) GO TO 30	00034370
C	PRINT NON-DEFINED MESSAGE	00034380
	IF (IEQDF.EQ.1) WRITE(IUNIT,50) IER,NAMEQ,IEQ,NAMUPK	00034390
	IF (IEQDF.EQ.0) WRITE(IUNIT,50) IER,NAMUPK	00034400
	30 IEQDF = 0	00034410
	RETURN	00034420
	35 FORMAT(19H *** TERMINAL ERROR,10X,7H(IER = ,I3,	00034430
	1 20H) FROM IMSL ROUTINE ,6A1,A1,6A1)	00034440
	40 FORMAT(27H *** WARNING WITH FIX ERROR,2X,7H(IER = ,I3,	00034450
	1 20H) FROM IMSL ROUTINE ,6A1,A1,6A1)	00034460
	45 FORMAT(18H *** WARNING ERROR,11X,7H(IER = ,I3,	00034470
	1 20H) FROM IMSL ROUTINE ,6A1,A1,6A1)	00034480
	50 FORMAT(20H *** UNDEFINED ERROR,9X,7H(IER = ,I5,	00034490
	1 20H) FROM IMSL ROUTINE ,6A1,A1,6A1)	00034500
	SAVE P FOR P = R CASE	00034510
	P IS THE PAGE NAMUPK	00034520
	R IS THE ROUTINE NAMUPK	00034530
	55 IEQDF = 1	00034540
	DO 60 I=1,6	00034550
	60 NAMEQ(I) = NAMUPK(I)	00034560
	65 RETURN	00034570
	END	00034580
		00034590
		00034600
		00034610
		00034620
	SUBROUTINE UGETIO(IOP1,NIN,NOUT)	00034630
	SPECIFICATIONS FOR ARGUMENTS	00034640
	INTEGER IOPT,NIN,NOUT	00034650
	SPECIFICATIONS FOR LOCAL VARIABLES	00034660
	INTEGER NIND,NOUTD	00034670
	DATA NIND/S/,NOUTD/6/	00034680
	FIRST EXECUTABLE STATEMENT	00034690
	IF (IOPT.EQ.3) GO TO 10	00034700
	IF (IOPT.EQ.2) GO TO 5	00034710
	IF (IOPT.NE.1) GO TO 9005	00034720
	NIN = NIND	00034730
	NOUT = NOUTD	00034740
	GO TO 9005	00034750

```

5 NIND = NIN
GO TO 9005
10 NOUTD = NOUT
9005 RETURN
END

```

C
C
C

SUBROUTINE VXADD(A,ACC)

C
C
C
C

DOUBLE PRECISION A,ACC(2)
DOUBLE PRECISION X,Y,Z,ZZ

SPECIFICATIONS FOR ARGUMENTS
SPECIFICATIONS FOR LOCAL VARIABLES
FIRST EXECUTABLE STATEMENT

X = ACC(1)
Y = A
IF (DABS(ACC(1)).GE.DABS(A)) GO TO 1
X = A
Y = ACC(1)

C

1 Z = X+Y
ZZ = (X-Z)+Y

COMPUTE Z+ZZ = ACC(1)+A EXACTLY

C

COMPUTE ZZ+ACC(2) USING DOUBLE
PRECISION ARITHMETIC

C

ZZ = ZZ+ACC(2)

COMPUTE ACC(1)+ACC(2) = Z+ZZ EXACTLY

C

ACC(1) = Z+ZZ
ACC(2) = (Z-ACC(1))+ZZ
RETURN
END

C

C

C

SUBROUTINE VXMUL (A,B,ACC)

C

DOUBLE PRECISION A,B,ACC(2)

SPECIFICATIONS FOR ARGUMENTS

C

SPECIFICATIONS FOR LOCAL VARIABLES

DOUBLE PRECISION X,HA,TA,HB,TB
INTEGER IX(2),I
LOGICAL L1
EQUIVALENCE (X,LX(1)),IX(1)), (I,LI(1))
DATA I/O

C

SPLIT A = HA+TA
B = HB+TB
FIRST EXECUTABLE STATEMENT

C

X = A
LI(4) = LX(5)
IX(2) = 0
I = (I/16)*16
LX(5) = LI(4)
HA=X
TA=A-HA
X = B
LI(4) = LX(5)
IX(2) = 0
I = (I/16)*16
LX(5) = LI(4)
HB = X
TB = B-HB

00034760
00034770
00034780
00034790
00034800
00034810
00034820
00034830
00034840
00034850
00034860
00034870
00034880
00034890
00034900
00034910
00034920
00034930
00034940
00034950
00034960
00034970
00034980
00034990
00035000
00035010
00035020
00035030
00035040
00035050
00035060
00035070
00035080
00035090
00035100
00035110
00035120
00035130
00035140
00035150
00035160
00035170
00035180
00035190
00035200
00035210
00035220
00035230
00035240
00035250
00035260
00035270
00035280
00035290
00035300
00035310
00035320
00035330
00035340
00035350

C		RADIX THE BASE OF THE FLOATING-POINT	00035960
C		NUMBER SYSTEM USED	00035970
	DATA	RINFP/27FFFFFFF/	00035980
	DATA	REPSP/200100000/	00035990
	DATA	RADIX/16.0/	00036000
	DATA	RE: 1/23410000000000000/	00036010
	DATA	ZERO/0.000/ONE/1.000/	00036020
C		ZRPOLY USES SINGLE PRECISION	00036030
C		CALCULATIONS FOR SCALING, BOUNDS	00036040
C		AND ERROR CALCULATIONS.	00036050
C		FIRST EXECUTABLE STATEMENT	00036060
	IER = 0		00036070
	IF (NDEG .GT. 100 .OR. NDEG .LT. 1) GO TO 165		00036080
	ETA = REPSP		00036090
	AME = ETA		00036100
	RMRE = ETA		00036110
	RLO = REPSP/ETA		00036120
C		INITIALIZATION OF CONSTANTS FOR	00036130
C		SHIFT ROTATION	00036140
	XX = .7071068		00036150
	YY = -XX		00036160
	SINR = .9975641		00036170
	COSR = -.06975647		00036180
	N = NDEG		00036190
	NN = N+1		00036200
C		ALGORITHM FAILS IF THE LEADING	00036210
C		COEFFICIENT IS ZERO.	00036220
	IF (A(1).NE.ZERO) GO TO 5		00036230
	IER = 130		00036240
	GO TO 9000		00036250
C		REMOVE THE ZEROS AT THE ORIGIN IF	00036260
C		ANY	00036270
	5 IF (A(NN).NE.ZERO) GO TO 10		00036280
	J = NDEG-N+1		00036290
	JJ = J+NDEG		00036300
	Z(J) = ZERO		00036310
	Z(JJ) = ZERO		00036320
	NN = NN-1		00036330
	N = N-1		00036340
	IF (NN.EQ.1) GO TO 9005		00036350
	GO TO 5		00036360
C		MAKE A COPY OF THE COEFFICIENTS	00036370
	10 DO 15 I=1,NN		00036380
	P(I) = A(I)		00036390
	15 CONTINUE		00036400
C		START THE ALGORITHM FOR ONE ZERO	00036410
	20 IF (N.GT.2) GO TO 30		00036420
	IF (N.LT.1) GO TO 9005		00036430
C		CALCULATE THE FINAL ZERO OR PAIR OF	00036440
C		ZEROS	00036450
	IF (N.EQ.2) GO TO 25		00036460
	Z(NDEG) = -P(2)/P(1)		00036470
	Z(NDEG+NDEG) = ZERO		00036480
	GO TO 145		00036490
	25 CALL ZRPOLI (P(1),P(2),P(3),Z(NDEG-1),Z(NDEG+NDEG-1),Z(NDEG),		00036500
	1 Z(NDEG+NDEG))		00036510
	GO TO 145		00036520
		FIND LARGEST AND SMALLEST MODULI OF	00036530
		COEFFICIENTS.	00036540
	30 RMAX = 0.		00036550

	RMIN = RINFP	00036560
	DO 35 I=1,NN	00036570
	X = ABS(SNOL(P(I)))	00036580
	IF (X.GT.RMAX) RMAX = X	00036590
	IF (X.NE.0. .AND.X.LT.RMIN) RMIN = X	00036600
35	CONTINUE	00036610
	SCALE IF THERE ARE LARGE OR VERY	00036620
	SMALL COEFFICIENTS COMPUTES A	00036630
	SCALE FACTOR TO MULTIPLY THE	00036640
	COEFFICIENTS OF THE POLYNOMIAL.	00036650
	THE SCALING IS DONE TO AVOID	00036660
	OVERFLOW AND TO AVOID UNDETECTED	00036670
	UNDERFLOW INTERFERING WITH THE	00036680
	CONVERGENCE CRITERION.	00036690
	THE FACTOR IS A POWER OF THE BASE	00036700
	SC = RLJ/RMIN	00036710
	IF (SC.GT.1.0) GO TO 40	00036720
	IF (RMAX.LT.10.) GO TO 55	00036730
	IF (SC.EQ.0.) SC = REPSHRRADIXXRADIX	00036740
	GO TO 45	00036750
40	IF (RINFP/SC.LT.RMAX) GO TO 55	00036760
45	L = ALOG(SC)/ALOG(RADIX)+.5	00036770
	IF (L.EQ. 0) GO TO 55	00036780
	FACTOR = DBLE(RADIX)**L	00036790
	DO 50 I=1,NN	00036800
50	P(I) = FACTOR**P(I)	00036810
	COMPUTE LOWER BOUND ON MODULI OF	00036820
	ZEROS.	00036830
55	DO 60 I=1,NN	00036840
60	PT(I) = ABS(SNOL(P(I)))	00036850
	PT(NN) = -PT(NN)	00036860
	COMPUTE UPPER ESTIMATE OF BOUND	00036870
	X = EXP((ALOG(-PT(NN))-ALOG(PT(1)))/N)	00036880
	IF (PT(N).EQ.0.) GO TO 65	00036890
	IF NEWTON STEP AT THE ORIGIN IS	00036900
	BETTER, USE IT.	00036910
	XM = -PT(NN)/PT(N)	00036920
	IF (XM.LT.X) X = XM	00036930
	CHOP THE INTERVAL (0,X) UNTIL FF.LE.0	00036940
65	XM = XM*.1	00036950
	FF = PT(1)	00036960
	DO 70 I=2,NN	00036970
70	FF = FF*XM+PT(I)	00036980
	IF (FF.LE.0.) GO TO 75	00036990
	X = XM	00037000
	GO TO 65	00037010
75	DX = X	00037020
	DO NEWTON ITERATION UNTIL X	00037030
	CONVERGES TO TWO DECIMAL PLACES	00037040
80	IF (ABS(DX/X).LE..005) GO TO 90	00037050
	FF = PT(1)	00037060
	DF = FF	00037070
	DO 85 I=2,N	00037080
	FF = FF*XM+PT(I)	00037090
	DF = DF*XM+FF	00037100
85	CONTINUE	00037110
	FF = FF*XM+PT(NN)	00037120
	DX = FF/DF	00037130
	X = X-DX	00037140
	GO TO 80	00037150

C	90 BND = X		00037160
C		COMPUTE THE DERIVATIVE AS THE INITIAL	00037170
C		K POLYNOMIAL AND DO 5 STEPS WITH	00037180
		NO SHIFT	00037190
	NM1 = N-1		00037200
	FN = ONE/N		00037210
	DO 95 I=2,N		00037220
	95 RK(I) = (NN-I)*P(I)*FN		00037230
	RK(1) = P(1)		00037240
	AA = P(NN)		00037250
	BB = P(N)		00037260
	ZEROK = RK(N).EQ.ZERO		00037270
	DO 115 JJ=1,5		00037280
	CC = RK(N)		00037290
	IF (ZEROK) GO TO 105		00037300
C		USE SCALED FORM OF RECURRENCE IF	00037310
C		VALUE OF K AT 0 IS NONZERO	00037320
	T = -AA/CC		00037330
	DO 100 I=1,NM1		00037340
	J = NN-I		00037350
	RK(J) = T*RK(J-1)+P(J)		00037360
100	CONTINUE		00037370
	RK(1) = P(1)		00037380
	ZEROK = DABS(RK(N)).LE.DABS(BB)*ETA*10.		00037390
	GO TO 115		00037400
C		USE UNSCALED FORM OF RECURRENCE	00037410
105	DO 110 I=1,NM1		00037420
	J = NN-I		00037430
	RK(J) = RK(J-1)		00037440
110	CONTINUE		00037450
	RK(1) = ZERO		00037460
	ZEROK = RK(N).EQ.ZERO		00037470
115	CONTINUE		00037480
C		SAVE X FOR RESTARTS WITH NEW SHIFTS	00037490
	DO 120 I=1,N		00037500
120	TEMP(I) = RK(I)		00037510
C		LOOP TO SELECT THE QUADRATIC	00037520
C		CORRESPONDING TO EACH NEW SHIFT	00037530
C	DO 140 ICNT=1,20		00037540
C		QUADRATIC CORRESPONDS TO A DOUBLE	00037550
C		SHIFT TO A NON-REAL POINT AND ITS	00037560
C		COMPLEX CONJUGATE. THE POINT HAS	00037570
C		MODULUS BND AND AMPLITUDE ROTATED	00037580
C		BY 94 DEGREES FROM THE PREVIOUS	00037590
C		SHIFT	00037600
	XXX = COSM*XX-SINR*YY		00037610
	YY = SINR*XX+COSR*YY		00037620
	XX = XXX		00037630
	SR = BND*XX		00037640
	SI = BND*YY		00037650
	U = -SR-SR		00037660
	V = BND*BND		00037670
C		SECOND STAGE CALCULATION, FIXED	00037680
C		QUADRATIC	00037690
	CALL ZRPOLB (20*ICNT,NZ)		00037700
	IF (NZ.EQ.0) GO TO 130		00037710
C		THE SECOND STAGE JUMPS DIRECTLY TO	00037720
C		ONE OF THE THIRD STAGE ITERATIONS	00037730
C		AND RETURNS HERE IF SUCCESSFUL.	00037740
C		DEFLATE THE POLYNOMIAL, STORE THE	00037750

C		ZERO OR ZEROS AND RETURN TO THE	00037760
C		MAIN ALGORITHM.	00037770
	J = NDEG-N+1		00037780
	JJ = J+NDEG		00037790
	Z(J) = SZR		00037800
	Z(JJ) = SZI		00037810
	NN = NN-NZ		03037820
	N = NN-1		00037830
	DO 125 I=1,NN		00037840
125	P(I) = QP(I)		00037850
	IF (NZ.EQ.1) GO TO 20		00037860
	Z(J+1) = RLZR		00037870
	Z(JJ+1) = RLZI		00037880
	GO TO 20		00037890
C		IF THE ITERATION IS UNSUCCESSFUL	00037900
C		ANOTHER QUADRATIC IS CHOSEN AFTER	00037910
C		RESTORING K	00037920
	130 DO 133 I=1,N		00037930
	133 RK(I) = TEMP(I)		00037940
	140 CONTINUE		00037950
C		RETURN WITH FAILURE IF NO	00037960
C		CONVERGENCE WITH 20 SHIFTS	00037970
C	IFR = 131	CONVERT ZEROS (Z) IN COMPLEX FORM	00037980
	145 DO 150 I=1,NDEG		00037990
	NPI = NDEG+I		00038000
	P(I) = Z(NPI)		00038010
	150 CONTINUE		00038020
	N2 = NDEG+NDEG		00038030
	J = NDEG		00038040
	DO 155 I=1,NDEG		00038050
	Z(N2-1) = Z(J)		00038060
	Z(N2) = P(J)		00038070
	N2 = N2-2		00038080
	J = J-1		00038090
	155 CONTINUE		00038100
	IF (IER.EQ. 0) GO TO 9005		00038110
C		SET UNFOUND ROOTS TO MACHINE INFINITY	00038120
	N2 = 2*(NDEG-NN)+3		00038130
	DO 160 I=1,N		00038140
	Z(N2) = RINFP		00038150
	Z(N2+1) = RINFP		00038160
	N2 = N2+2		00038170
	160 CONTINUE		00038180
	GO TO 9000		00038190
	165 IER = 129		00038200
	9000 CONTINUE		00038210
	CALL UERTST (IER,6HZRPOLY)		00038220
	9005 RETURN		00038230
	END		00038240
C			00038250
C			00038260
C			00038270
C			00038280
C	SUBROUTINE ZRPQLB (L2,NZ)		00038290
C	INTEGER	SPECIFICATIONS FOR ARGUMENTS	00038300
C	L2,NZ		00038310
C	INTEGER	SPECIFICATIONS FOR LOCAL VARIABLES	00038320
C	REAL		00038330
1		N,NN,J,ITYPE,I,IFLAG	00038340
		ARE,BETAS,BETAV,ETA,GSS,OTS,OTV,OVV,RMRE,SS,	00038350
		TS,TSS,TV,TVV,VV	

	DOUBLE PRECISION	P(101),QP(101),RK(101),QK(101),SVK(101)	00038360
	DOUBLE PRECISION	SR,S1,U,V,RA,RD,C,D,A1,A2,A3,	00038370
1		A6,A7,E,F,O,H,SZR,SZI,RLZR,RLZI,	00038380
2		SVU,SVV,UI,VI,S,ZERO	00038390
	LOGICAL	VPASS,SPASS,VTRY,STRY	00038400
	COMMON /ZRPQLJ/	P,QP,RK,QK,SVK,SR,S1,U,V,RA,RD,C,D,A1,A2,A3,A6,	00038410
1		A7,E,F,O,H,SZR,SZI,RLZR,RLZI,ETA,ARE,RMRE,N,NH	00038420
	DATA	ZERO/0.000/	00038430
C	NZ = 0	FIRST EXECUTABLE STATEMENT	00038440
C		COMPUTES UP TO L2 FIXED SHIFT	00038450
C		K-POLYNOMIALS, TESTING FOR	00038460
C		CONVERGENCE IN THE LINEAR OR	00038470
C		QUADRATIC CASE. INITIATES ONE OF	00038480
C		THE VARIABLE SHIFT ITERATIONS AND	00038490
C		RETURNS WITH THE NUMBER OF ZEROS	00038500
C		FOUND.	00038510
C		L2 - LIMIT OF FIXED SHIFT STEPS	00038520
C		NZ -NUMBER OF ZEROS FOUND	00038530
	BETAV = .25		00038540
	BETAS = .25		00038550
	OSS = SR		00038560
	OVV = V		00038570
C		EVALUATE POLYNOMIAL BY SYNTHETIC	00038580
C		DIVISION	00038590
	CALL ZRPQLH (NH,U,V,P,QP,RA,RD)		00038600
	CALL ZRPQLE (ITYPE)		00038610
	DO 40 J=1,L2		00038620
C		CALCULATE NEXT K POLYNOMIAL AND	00038630
C		ESTIMATE V	00038640
	CALL ZRPQLF (ITYPE)		00038650
	CALL ZRPQLE (ITYPE)		00038660
	CALL ZRPQLO (ITYPE,UI,VI)		00038670
	VV = VI		00038680
C		ESTIMATE S	00038690
	SS = 0.		00038700
	IF (RK(N).NE.ZERO) SS = -P(NN)/RK(N)		00038710
	TV = 1.		00038720
	TS = 1.		00038730
	IF (J.EQ.1.OR.ITYPE.EQ.3) GO TO 35		00038740
C		COMPUTE RELATIVE MEASURES OF	00038750
C		CONVERGENCE OF S AND V SEQUENCES	00038760
	IF (VV.NE.0.) TV = ABS((VV-OVV)/VV)		00038770
	IF (SS.NE.0.) TS = ABS((SS-USS)/SS)		00038780
C		IF DECREASING, MULTIPLY TWO MOST	00038790
C		RECENT CONVERGENCE MEASURES	00038800
	TVV = 1.		00038810
	IF (TV.LT.OTV) TVV = TV*OTV		00038820
	TSS = 1.		00038830
	IF (TS.LT.OTS) TSS = TS*OTS		00038840
C		COMPARE WITH CONVERGENCE CRITERIA	00038850
	VPASS = TVV.LT.BETAV		00038860
	SPASS = TSS.LT.BETAS		00038870
	IF (.NOT.(SPASS.OR.VPASS)) GO TO 35		00038880
C		AT LEAST ONE SEQUENCE HAS PASSED THE	00038890
C		CONVERGENCE TEST. STORE VARIABLES	00038900
C		BEFORE ITERATING	00038910
	SVU = U		00038920
	SVV = V		00038930
	DO 5 I=1,N		00038940
			00038950

5	SVK(I) = RK(I)		00038960
	S = SS		00038970
C		CHOOSE ITERATION ACCORDING TO THE	00038980
C		FASTEST CONVERGING SEQUENCE	00038990
	VTRY = .FALSE.		00039000
	STRY = .FALSE.		00039010
10	IF (SPASS.AND.((.NOT.VPASS).OR.TSS.LT.TVV)) GO TO 20		00039020
	CALL ZRPQLC (UI,VI,NZ)		00039030
	IF (NZ.GT.0) RETURN		00039040
C		QUADRATIC ITERATION HAS FAILED. FLAG	00039050
C		THAT IT HAS BEEN TRIED AND	00039060
C		DECREASE THE CONVERGENCE	00039070
		CRITERION.	00039080
	VTRY = .TRUE.		00039090
	BETAV = BETAV*.25		00039100
C		TRY LINEAR ITERATION IF IT HAS NOT	00039110
C		BEEN TRIED AND THE S SEQUENCE IS	00039120
C		CONVERGING	00039130
	IF (STRY.OR.((.NOT.SPASS)) GO TO 25		00039140
	DO 15 I=1,N		00039150
15	RK(I) = SVK(I)		00039160
20	CALL ZRPQLD (S,NZ,IFLAG)		00039170
	IF (NZ.GT.0) RETURN		00039180
C		LINEAR ITERATION HAS FAILED. FLAG	00039190
C		THAT IT HAS BEEN TRIED AND	00039200
		DECREASE THE CONVERGENCE CRITERION	00039210
	STRY = .TRUE.		00039220
	BETAS = BETAS*.25		00039230
	IF (IFLAG.EQ.0) GO TO 25		00039240
C		IF LINEAR ITERATION SIGNALS AN	00039250
C		ALMOST DOUBLE REAL ZERO ATTEMPT	00039260
C		QUADRATIC ITERATION	00039270
	UI = -(S+S)		00039280
	VI = S*S		00039290
	GO TO 10		00039300
C		RESTORE VARIABLES	00039310
25	U = SVU		00039320
	V = SVV		00039330
	DO 30 I=1,N		00039340
30	RK(I) = SVK(I)		00039350
C		TRY QUADRATIC ITERATION IF IT HAS	00039360
C		NOT BEEN TRIED AND THE V SEQUENCE	00039370
C		IS CONVERGING	00039380
	IF (VPASS.AND.((.NOT.VTRY)) GO TO 10		00039390
C		RECOMPUTE QP AND SCALAR VALUES TO	00039400
C		CONTINUE THE SECOND STAGE	00039410
	CALL ZRPQLH (NN,U,V,P,QP,RA,RB)		00039420
	CALL ZRPQLE (ITYPE)		00039430
35	QVV = VV		00039440
	QSS = SS		00039450
	QTV = TV		00039460
	QTS = TS		00039470
40	CONTINUE		00039480
	RETURN		00039490
	END		00039500
C			00039510
C			00039520
C			00039530
	SUBROUTINE ZRPQLC (UU,VV,NZ)		00039540
C		SPECIFICATIONS FOR ARGUMENTS	00039550

	INTEGER	NZ	00039560
	DOUBLE PRECISION	UU,VV	00039570
C		SPECIFICATIONS FOR LOCAL VARIABLES	00039580
	INTEGER	N,NN,J,I,ITYPE	00039590
	REAL	ARE,EE,ETA,OMP,RELSTP,RMP,RMRE,T,ZM	00039600
	DOUBLE PRECISION	P(101),QP(101),RK(101),QK(101),SVK(101)	00039610
	DOUBLE PRECISION	SR,SI,U,V,RA,RE,C,D,A1,A2,A3,	00039620
1		A6,A7,E,F,O,H,SZR,SZI,RLZR,RLZI,	00039630
2		UI,VI,ZERO,PT01,ONE	00039640
	LOGICAL	TRIED	00039650
	COMMON /ZRPQLJ/	P,QP,RK,QK,SVK,SR,SI,U,V,RA,RE,C,D,A1,A2,A3,A6,	00039660
1		A7,E,F,O,H,SZR,SZI,RLZR,RLZI,ETA,ARE,RMRE,N,NN	00039670
	DATA	ZERO,PT01,ONE/0.000,0.0100,1.000/	00039680
C		FIRST EXECUTABLE STATEMENT	00039690
	NZ = 0		00039700
C		VARIABLE-SHIFT K-POLYNOMIAL	00039710
C		ITERATION FOR A QUADRATIC FACTOR	00039720
C		CONVERGES ONLY IF THE ZEROS ARE	00039730
C		EQUIMODULAR OR NEARLY SO	00039740
C		UU,VV - COEFFICIENTS OF STARTING	00039750
C		QUADRATIC	00039760
		NZ - NUMBER OF ZERO FOUND	00039770
	TRIED = .FALSE.		00039780
	U = UU		00039790
	V = VV		00039800
	J = 0		00039810
C		MAIN LOOP	00039820
C	3	CALL ZRPQLI (ONE,U,V,SZR,SZI,RLZR,RLZI)	00039830
C		RETURN IF ROOTS OF THE QUADRATIC ARE	00039840
C		REAL AND NOT CLOSE TO MULTIPLE OR	00039850
C		NEARLY EQUAL AND OF OPPOSITE SIGN	00039860
C		IF (DABS(DABS(SZR)-DABS(RLZR)).GT.PT01*DABS(RLZR)) RETURN	00039870
C		EVALUATE POLYNOMIAL BY QUADRATIC	00039880
C		SYNTHETIC DIVISION	00039890
	CALL ZRPQLH (NN,U,V,P,QP,RA,RE)		00039900
	RMP = DABS(RA-SZR*RB)+DABS(SZIRB)		00039910
C		COMPUTE A RIGOROUS BOUND ON THE	00039920
C		ROUNDING ERROR IN EVALUATING P	00039930
	ZM = SQRT(ABS(SNOL(V)))		00039940
	EE = 2.*ABS(SNOL(QP(1)))		00039950
	T = -SZR*RB		00039960
	DO 10 I=2,N		00039970
	10 EE = EE*ZM+ABS(SNOL(QP(I)))		00039980
	EE = EE*ZM+ABS(SNOL(RA)+T)		00039990
	EE = (5.*RMRE+4.*ARE)*EE-(5.*RMRE+2.*ARE)*ABS(SNOL(RA)+T)+		00040000
1	ABS(SNOL(RB))*ZM)+2.*ARE*ABS(T)		00040010
C		ITERATION HAS CONVERGED SUFFICIENTLY	00040020
C		IF THE POLYNOMIAL VALUE IS LESS	00040030
C		THAN 20 TIMES THIS BOUND	00040040
	IF (RMP.GT.20.*EE) GO TO 15		00040050
	NZ = NZ + 1		00040060
	RETURN		00040070
15	J = J+1		00040080
C		STOP ITERATION AFTER 20 STEPS	00040090
	IF (J.GT.20) RETURN		00040100
	IF (J.LT.2) GO TO 25		00040110
	IF (RELSTP.GT..01.OR.RMP.LT.OMP.OR.TRIED) GO TO 25		00040120
C		A CLUSTER APPEARS TO BE STALLING THE	00040130
C		CONVERGENCE. FIVE FIXED SHIFT	00040140
C		STEPS ARE TAKEN WITH A U,V CLOSE	00040150

C	IF (RELSTP.LT.ETA) RELSTP = ETA	00040160
	RELSTP = SQRT(RELSTP)	00040170
	U = U-UXRELSTP	00040180
	V = V+VXRELSTP	00040190
	CALL ZRPQLH (NN,U,V,P,QP,RA,RB)	00040200
	DO 20 I=1,5	00040210
	CALL ZRPQLE (ITYPE)	00040220
	CALL ZRPQLF (ITYPE)	00040230
20	CONTINUE	00040240
	TRIED = .TRUE.	00040250
	J = 0	00040260
25	OMP = RMP	00040270
C		00040280
C	CALCULATE NEXT K POLYNOMIAL AND NEW U AND V	00040290
	CALL ZRPQLE (ITYPE)	00040300
	CALL ZRPQLF (ITYPE)	00040310
	CALL ZRPQLE (ITYPE)	00040320
	CALL ZRPQLO (ITYPE,UI,VI)	00040330
C		00040340
C	IF VI IS ZERO THE ITERATION IS NOT CONVERGING	00040350
	IF (VI.EQ.ZERO) RETURN	00040360
	RELSTP = DABS((VI-V)/VI)	00040370
	U = UI	00040380
	V = VI	00040390
	GO TO 5	00040400
	END	00040410
C		00040420
C		00040430
C		00040440
C	SUBROUTINE ZRPQLD (SSS,NZ,IFLAG)	00040450
C	INTEGER NZ,IFLAG	00040460
	DOUBLE PRECISION SSS	00040470
C		00040480
C	INTEGER N,NN,J,X	00040490
	REAL ARE,EE,ETA,OMP,RMP,RMS,RMRE	00040500
	DOUBLE PRECISION P(101),QP(101),RK(101),QK(101),SVK(101)	00040510
	DOUBLE PRECISION SR,SI,U,V,RA,RB,C,D,A1,A2,A3,	00040520
1	A6,A7,E,F,G,H,SZR,SZI,RLZR,RLZI,	00040530
2	PV,RKV,T,S,ZERO,PT001	00040540
COMMON /ZRPQLJ/	P,QP,RK,QK,SVK,SR,SI,U,V,RA,RB,C,D,A1,A2,A3,A6,	00040550
1	A7,E,F,G,H,SZR,SZI,RLZR,RLZI,ETA,ARE,RMRE,N,NN	00040560
	ZERO/0.000/,PT001/0.00100/	00040570
C		00040580
C	VARIABLE-SHIFT M POLYNOMIAL	00040590
C	ITERATION FOR A REAL ZERO SSS -	00040600
C	STARTING ITERATE	00040610
C	NZ - NUMBER OF ZERO FOUND	00040620
C	IFLAG - FLAG TO INDICATE A PAIR OF	00040630
C	ZEROS NEAR REAL AXIS	00040640
C	FIRST EXECUTABLE STATEMENT	00040650
	NZ = 0	00040660
	S = SSS	00040670
	IFLAG = 0	00040680
	J = 0	00040690
C		00040700
C	5 PV = P(1)	00040710
	EVALUATE P AT S	00040720
	QP(1) = PV	00040730
	DO 10 I=7,NN	00040740
		00040750

	PV = PV*S+P(I)	00040760
	QP(I) = PV	00040770
10	CONTINUE	00040780
	RMP = DABS(PV)	00040790
C		00040800
C	COMPUTE A RIGOROUS BOUND ON THE	00040810
	ERROR IN EVALUATING P	00040820
	RMS = DABS(S)	00040830
	EE = (RMRE/(ARE+RMRE))*ABS(SNGL(QP(1)))	00040840
	DO 15 I=2,N	00040850
15	EE = EE*RMS+ABS(SNGL(QP(I)))	00040860
C		00040870
C	ITERATION HAS CONVERGED SUFFICIENTLY	00040880
	IF THE POLYNOMIAL VALUE IS LESS	00040890
	THAN 20 TIMES THIS BOUND	00040900
	IF (RMP.GT.20.*((ARE+RMRE)*EE-RMRE*RMP)) GO TO 20	00040910
	NZ = 1	00040920
	SZR = S	00040930
	SZI = ZERO	00040940
	RETURN	00040950
20	J = J+1	00040960
C		00040970
	STOP ITERATION AFTER 10 STEPS	00040980
	IF (J.GT.10) RETURN	00040990
	IF (J.LT.2) GO TO 25	00041000
	IF (DABS(T).GT.PT001*DABS(S-T).OR.RMP.LE.OMP) GO TO 25	00041010
C		00041020
C	A CLUSTER OF ZEROS NEAR THE REAL	00041030
C	AXIS HAS BEEN ENCOUNTERED RETURN	00041040
C	WITH IFLAG SET TO INITIATE A	00041050
	QUADRATIC ITERATION	00041060
	IFLAG = 1	00041070
	SSS = S	00041080
	RETURN	00041090
C		00041100
C	RETURN IF THE POLYNOMIAL VALUE HAS	00041110
	INCREASED SIGNIFICANTLY	00041120
25	OMP = RMP	00041130
C		00041140
C	COMPUTE T, THE NEXT POLYNOMIAL, AND	00041150
	THE NEW ITERATE	00041160
	RKV = RK(1)	00041170
	QK(1) = RKV	00041180
	DO 30 I=2,N	00041190
	RKV = RKV*S+RK(I)	00041200
	QK(I) = RKV	00041210
30	CONTINUE	00041220
	IF (DABS(RKV).LE.DABS(RK(N))*10.*ETA) GO TO 40	00041230
C		00041240
C	USE THE SCALED FORM OF THE	00041250
C	RECURRENCE IF THE VALUE OF K AT S	00041260
	IS NONZERO	00041270
	T = -PV/RKV	00041280
	RK(1) = QP(1)	00041290
	DO 35 I=2,N	00041300
35	RK(I) = T*QK(I-1)+QP(I)	00041310
	GO TO 20	00041320
C		00041330
	USE UNSCALED FORM	00041340
40	RK(1) = ZERO	00041350
	DO 45 I=2,N	
45	RK(I) = QK(I-1)	
50	RKV = RK(1)	
	DO 55 I=2,N	
55	RKV = RKV*S+RK(I)	
	T = ZERO	
	IF (DABS(RKV).GT.DABS(RK(N))*10.*ETA) T = -PV/RKV	
	S = S+T	

GO TO 5		00041360
END		00041370
		00041380
		00041390
		00041400
IMSL ROUTINE NAME - ZRPQLE		00041410
		00041420
		00041430
COMPUTER - IBM/DOUBLE		00041440
		00041450
LATEST REVISION - JANUARY 1, 1978		00041460
		00041470
		00041480
SUBROUTINE ZRPQLE (ITYPE)		00041490
	SPECIFICATIONS FOR ARGUMENTS	00041500
INTEGER ITYPE		00041510
	SPECIFICATIONS FOR LOCAL VARIABLES	00041520
INTEGER N,NN		00041530
REAL ARE,ETA,FMRE		00041540
DOUBLE PRECISION P(101),QF(101),RK(101),QK(101),SVK(101)		00041550
DOUBLE PRECISION SR,SI,U,V,RA,RB,C,D,A1,A2,A3,		00041560
1 A6,A7,E,F,G,H,SZR,SZI,RLZR,RLZI		00041570
1 COMMON /ZRPQLJ/ P,QP,RK,QK,SVK,SR,SI,U,V,RA,RB,C,D,A1,A2,A3,A6,		00041580
1 A7,E,F,G,H,SZR,SZI,RLZR,RLZI,ETA,ARE,FMRE,N,NN		00041590
	THIS ROUTINE CALCULATES SCALAR	00041600
	QUANTITIES USED TO COMPUTE THE	00041610
	NEXT K POLYNOMIAL AND NEW	00041620
	ESTIMATES OF THE QUADRATIC	00041630
	COEFFICIENTS	00041640
	ITYPE - INTEGER VARIABLE SET HERE	00041650
	INDICATING HOW THE CALCULATIONS	00041660
	ARE NORMALIZED TO AVOID OVERFLOW	00041670
	SYNTHETIC DIVISION OF K BY THE	00041680
	QUADRATIC 1,U,V	00041690
	FIRST EXECUTABLE STATEMENT	00041700
CALL ZRPQLH (N,U,V,RK,QK,C,D)		00041710
IF (DABS(C).GT.DABS(RK(N))*100.*ETA) GO TO 5		00041720
IF (DABS(D).GT.DABS(RK(N-1))*100.*ETA) GO TO 5		00041730
ITYPE = 3		00041740
	TYPE=3 INDICATES THE QUADRATIC IS	00041750
	ALMOST A FACTOR OF K	00041760
RETURN		00041770
5 IF (DABS(D).LT.DABS(C)) GO TO 10		00041780
ITYPE = 2		00041790
	TYPE=2 INDICATES THAT ALL FORMULAS	00041800
	ARE DIVIDED BY D	00041810
F = RA/D		00041820
F = C/D		00041830
G = U*RB		00041840
H = V*RB		00041850
A3 = (RA+G)*E+H*(RB/D)		00041860
A1 = RB*F-RA		00041870
A7 = (F+U)*RA+H		00041880
RETURN		00041890
10 ITYPE = 1		00041900
	TYPE=1 INDICATES THAT ALL FORMULAS	00041910
	ARE DIVIDED BY C	00041920
E = RA/C		00041930
F = D/C		00041940
G = U*E		00041950

	H = VMRB	00041960
	A5 = RA*E+(H/C+G)*RB	00041970
	A1 = RB-RA*(D/C)	00041980
	A7 = RA+G*D+H*F	00041990
	RETURN	00042000
	END	00042010
C		00042020
C		00042030
C	SUBROUTINE ZRPQLF (ITYPE)	00042040
C	INTEGER	00042050
C	ITYPE	00042060
	SPECIFICATIONS FOR ARGUMENTS	00042070
	SPECIFICATIONS FOR LOCAL VARIABLES	00042080
	INTEGER N,NN,I	00042090
	REAL ARE,ETA,RMRE	00042100
	DOUBLE PRECISION P(101),QP(101),RK(101),QK(101),SVK(101)	00042110
	DOUBLE PRECISION SR,SI,U,V,RA,RB,C,D,A1,A2,A3,	00042120
1	A6,A7,E,F,G,H,SZR,SZI,RLZR,RLZI,TEMP,ZERO	00042130
1	COMMON /ZRPQLJ/	00042140
	P,QP,RK,QK,SVK,SR,SI,U,V,RA,RB,C,D,A1,A2,A3,A6,	00042150
	A7,E,F,G,H,SZR,SZI,RLZR,RLZI,ETA,ARE,RMRE,N,NN	00042160
	ZERO/0.0D0/	00042170
C	COMPUTES THE NEXT K POLYNOMIALS	00042180
C	USING SCALARS COMPUTED IN ZRPQLE	00042190
	FIRST EXECUTABLE STATEMENT	00042200
	IF (ITYPE.EQ.3) GO TO 20	00042210
	TEMP = RA	00042220
	IF (ITYPE.EQ.1) TEMP = RB	00042230
	IF (DABS(A1).GT.DABS(TEMP)*ETA*10.) GO TO 10	00042240
C	IF A1 IS NEARLY ZERO THEN USE A	00042250
C	SPECIAL FORM OF THE RECURRENCE	00042260
	RK(1) = ZERO	00042270
	RK(2) = -A7*QP(1)	00042280
	DO 5 I=3,N	00042290
5	RK(I) = A3*QK(I-2)-A7*QP(I-1)	00042300
	RETURN	00042310
C	USE SCALED FORM OF THE RECURRENCE	00042320
10	A7 = A7/A1	00042330
	A3 = A3/A1	00042340
	RK(1) = QP(1)	00042350
	RK(2) = QP(2)-A7*QP(1)	00042360
	DO 15 I=3,N	00042370
15	RK(I) = A3*QK(I-2)-A7*QP(I-1)+QP(I)	00042380
	RETURN	00042390
C	USE UNSCALED FORM OF THE RECURRENCE	00042400
C	IF TYPE IS 3	00042410
20	RK(1) = ZERO	00042420
	RK(2) = ZERO	00042430
	DO 25 I=3,N	00042440
25	RK(I) = QK(I-2)	00042450
	RETURN	00042460
	END	00042470
C		00042480
C		00042490
C	IMSL ROUTINE NAME - ZRPQLG	00042500
C		00042510
C		00042520
C		00042530
C	COMPUTER - IBM/DOUBLE	00042540
C		00042550

```

C   LATEST REVISION      - JANUARY 1, 1978                                00042560
C                                                                    00042570
C   SUBROUTINE ZRPQLG (ITYPE,UU,VV)                                00042580
C                               SPECIFICATIONS FOR ARGUMENTS          00042590
C   INTEGER                    ITYPE                                00042600
C   DOUBLE PRECISION          UU,VV                                00042610
C                               SPECIFICATIONS FOR LOCAL VARIABLES    00042620
C   INTEGER                    N,NN                                00042630
C   REAL                      ARE,ETA,RMRE                        00042640
C   DOUBLE PRECISION          P(101),QP(101),RK(101),QK(101),SVK(101) 00042650
C   DOUBLE PRECISION          SR,SI,U,V,RA,RB,C,D,A1,A2,A3,      00042660
C                               A6,A7,E,F,G,H,SZR,SZI,RLZR,RLZI,    00042670
C                               A4,A5,B1,B2,C1,C2,C3,C4,TEMP,ZERO 00042680
C   1 COMMON /ZRPQLJ/        P,GP,RK,QK,SVK,SR,SI,U,V,RA,RB,C,D,A1,A2,A3,A6, 00042690
C   2                        A7,E,F,G,H,SZR,SZI,RLZR,RLZI,ETA,ARE,RMRE,N,NN 00042700
C   1 DATA                  ZERO/0.0D0/                          00042710
C                                                                    00042720
C                               COMPUTE NEW ESTIMATES OF THE        00042730
C                               QUADRATIC COEFFICIENTS USING THE    00042740
C                               SCALARS COMPUTED IN ZRPQLG          00042750
C                               USE FORMULAS APPROPRIATE TO SETTING 00042760
C                               OF TYPE.                            00042770
C                               FIRST EXECUTABLE STATEMENT          00042780
C                                                                    00042790
C   IF (ITYPE.EQ.3) GO TO 15                                         00042800
C   IF (ITYPE.EQ.2) GO TO 5                                           00042810
C   A4 = RA+UMRB+H*F                                                  00042820
C   A5 = C+(U+V*F)*D                                                  00042830
C   GO TO 10                                                           00042840
C   5 A4 = (RA+G)*F+H                                                  00042850
C   A5 = (F+U)*C+V*D                                                  00042860
C                                                                    00042870
C                               EVALUATE NEW QUADRATIC COEFFICIENTS. 00042880
C                                                                    00042890
C   10 B1 = -RK(N)/P(NN)                                              00042900
C   B2 = -(RK(N-1)+B1*P(N))/P(NN)                                    00042910
C   C1 = V*B2*A1                                                      00042920
C   C2 = B1*A7                                                         00042930
C   C3 = B1*B1*A3                                                      00042940
C   C4 = C1-C2-C3                                                      00042950
C   TEMP = A5+B1*A4-C4                                                 00042960
C   IF (TEMP.EQ.ZERO) GO TO 15                                         00042970
C   UU = U-(U*(C3+C2)+V*(B1*A1+B2*A7))/TEMP                          00042980
C   VV = V*(1+C4/TEMP)                                                 00042990
C   RETURN                                                             00043000
C                               IF TYPE=3 THE QUADRATIC IS ZEROED    00043010
C   15 UU = ZERO                                                       00043020
C   VV = ZERO                                                         00043030
C   RETURN                                                             00043040
C   END                                                                00043050
C                                                                    00043060
C   SUBROUTINE ZRPQLH (NN,U,V,P,Q,RA,RB)                            00043070
C                               SPECIFICATIONS FOR ARGUMENTS          00043080
C   INTEGER                    NN                                00043090
C   DOUBLE PRECISION          P(NN),Q(NN),U,V,RA,RB              00043100
C                               SPECIFICATIONS FOR LOCAL VARIABLES    00043110
C   INTEGER                    I                                00043120
C   DOUBLE PRECISION          C                                00043130
C                                                                    00043140
C                               DIVIDES P BY THE QUADRATIC 1,U,V     00043150
C                               PLACING THE QUOTIENT IN Q AND THE    00043160
C                               REMAINDER IN A,B

```

C	RB = P(1)	FIRST EXECUTABLE STATEMENT	00043160
	Q(1) = RB		00043170
	RA = P(2)-UMRB		00043180
	Q(2) = RA		00043190
	DO 5 I=3,NH		00043200
	C = P(I)-UMRA-VMRB		00043210
	Q(I) = C		00043220
	RB = RA		00043230
	RA = C		00043240
	5 CONTINUE		00043250
	RETURN		00043260
	END		00043270
C			00043280
C			00043290
C	IMSL ROUTINE NAME - ZRPQLI		00043300
C			00043310
C			00043320
C			00043330
C			00043340
C	COMPUTER - IBM/DOUBLE		00043350
C			00043360
C	LATEST REVISION - JANUARY 1, 1978		00043370
C			00043380
C	SUBROUTINE ZRPQLI (RA,B1,C,SR,SI,RLR,RLI)		00043390
C		SPECIFICATIONS FOR ARGUMENTS	00043400
C	DOUBLE PRECISION RA,B1,C,SR,SI,RLR,RLI		00043410
C		SPECIFICATIONS FOR LOCAL VARIABLES	00043420
C	DOUBLE PRECISION RB,D,E,ZERO,ONE,TWO		00043430
C	DATA ZERO,ONE,TWO/0.000,1.000,2.000/		00043440
C		CALCULATE THE ZEROS OF THE QUADRATIC	00043450
C		ANZMX2 + B1XZ + C. THE QUADRATIC	00043460
C		FORMULA, MODIFIED TO AVOID	00043470
C		OVERFLOW, IS USED TO FIND THE	00043480
C		LARGER ZERO IF THE ZEROS ARE REAL	00043490
C		AND BOTH ZEROS ARE COMPLEX.	00043500
C		THE SMALLER REAL ZERO IS FOUND	00043510
C		DIRECTLY FROM THE PRODUCT OF THE	00043520
C		ZEROS C/A	00043530
C		FIRST EXECUTABLE STATEMENT	00043540
C			00043550
C	IF (RA.NE.ZERO) GO TO 10		00043560
C	SR = ZERO		00043570
C	IF (B1.NE.ZERO) SR = -C/B1		00043580
C	RLR = ZERO		00043590
C	5 SI = ZERO		00043600
C	RLI = ZERO		00043610
C	RETURN		00043620
C	10 IF (C.NE.ZERO) GO TO 15		00043630
C	SR = ZERO		00043640
C	RLR = -B1/RA		00043650
C	GO TO 5		00043660
C		COMPUTE DISCRIMINANT AVOIDING	00043670
C		OVERFLOW	00043680
C	15 RB = B1/TWO		00043690
C	IF (DABS(RB).LT.DABS(C)) GO TO 20		00043700
C	E = ONE-(RA/RB)*(C/RB)		00043710
C	D = DSQRT(DABS(E))*DABS(RB)		00043720
C	GO TO 25		00043730
C	20 E = RA		00043740
C	IF (C.LT.?ERO) E = -RA		00043750

	E = RB/(RB/DABS(C))-E	00043760
	D = DSQRT(DABS(E))*DSQRT(DABS(C))	00043770
C	25 IF (E.LT.ZERO) GO TO 30	00043780
	REAL ZEROS	00043790
	IF (RB.GE.ZERO) D = -D	00043800
	RLR = (-RB+D)/RA	00043810
	SR = ZERO	00043820
	IF (RLR.NE.ZERO) SR = (C/RLR)/RA	00043830
	GO TO 5	00043840
C	COMPLEX CONJUGATE ZEROS	00043850
C	30 SR = -RB/RA	00043860
	RLR = SR	00043870
	SI = DABS(D/RA)	00043880
	RLI = -SI	00043890
	RETURN	00043900
	END	00043910
C		00043920
C		00043930
C		00043940
C	SUBROUTINE LEQ2C (A,N,IA,B,M,IB,IJOB,WA,WK,IER)	00043950
	COMPLEX*16 A(IA,1),B(IB,1),WA(N,1),TEMPA,TEMPC,TEMPC	00043960
	DOUBLE PRECISION WK(N),TA(2),TB(2),TC(2)	00043970
	DOUBLE PRECISION AR,AI,BR,BI,CR,CI,DXNORM,XNORM,ZERO	00043980
	DOUBLE PRECISION ACC(2)	00043990
	EQUIVALENCE (TA(1),TEMPA),(TB(1),TEMPB),(TC(1),TEMPC),	00044000
	(TA(1),AR),(TA(2),AI),(TB(1),BR),(TB(2),BI),	00044010
	(TC(1),CR),(TC(2),CI)	00044020
	DATA ZERO/0.0D0/	00044030
	DATA ITMAX/50/	00044040
C	FIRST EXECUTABLE STATEMENT	00044050
	IER = 0	00044060
	N1 = N+1	00044070
	N2 = N+2	00044080
	IF (IJOB .EQ. 2) GO TO 15	00044090
C	SAVE MATRIX A	00044100
	DO 10 I = 1,N	00044110
	DO 5 J = 1,N	00044120
	WA(I,J) = A(I,J)	00044130
5	CONTINUE	00044140
10	CONTINUE	00044150
C	FACTOR MATRIX A	00044160
	CALL LEQ1C (WA,N,N,B,M,IB,1,WK,IER)	00044170
	IF (IER .NE. 0) GO TO 9000	00044180
	IF (IJOB .EQ. 1) GO TO 9005	00044190
C	SAVE THE RIGHT HAND SIDES	00044200
15	DO 65 J = 1,M	00044210
	DO 20 I = 1,N	00044220
	WA(I,N1) = B(I,J)	00044230
20	CONTINUE	00044240
C	OBTAIN A SOLUTION	00044250
	CALL LEQ1C(WA,N,N,WA(1,N1),1,N,2,WK,IER)	00044260
C	COMPUTE THE NORM OF THE SOLUTION	00044270
	XNORM = ZERO	00044280
	DO 25 I = 1,N	00044290
	TEMPA = WA(I,N1)	00044300
	XNORM = DMAX1(XNORM,DABS(AR),DABS(AI))	00044310
25	CONTINUE	00044320
	IF (XNORM .EQ. ZERO) GO TO 65	00044330
C	COMPUTE RESIDUALS	00044340
		00044350

DO 50 ITER = 1, ITMAX	00044360
DO 40 I = 1, N	00044370
TEMPB = B(I, J)	00044380
ACC(1) = 0.000	00044390
ACC(2) = 0.000	00044400
CALL VXADD(BR, ACC)	00044410
DO 30 JJ = 1, N	00044420
TEMPA = A(I, JJ)	00044430
TEMPB = WA(JJ, N1)	00044440
CALL VXMUL(-AR, BR, ACC)	00044450
CALL VXMUL(AI, BI, ACC)	00044460
30 CONTINUE	00044470
CALL VXSTO(ACC, CR)	00044480
TEMPB = B(I, J)	00044490
ACC(1) = 0.000	00044500
ACC(2) = 0.000	00044510
CALL VXADD(BI, ACC)	00044520
DO 35 JJ = 1, N	00044530
TEMPA = A(I, JJ)	00044540
TEMPB = WA(JJ, N1)	00044550
CALL VXMUL(-AR, BI, ACC)	00044560
CALL VXMUL(-BR, AI, ACC)	00044570
35 CONTINUE	00044580
CALL VXSTO(ACC, CI)	00044590
WA(I, N2) = TEMPC	00044600
40 CONTINUE	00044610
CALL LEQTC(WA, N, N, WA(1, N2), 1, N, 2, WK, IER)	00044620
DXNORM = ZERO	00044630
C UPDATE THE SOLUTION	00044640
DO 45 I = 1, N	00044650
WA(I, N1) = WA(I, N1) + WA(I, N2)	00044660
TEMPA = WA(I, N2)	00044670
DXNORM = DMAX1(DXNORM, DABS(AR), DABS(AI))	00044680
45 CONTINUE	00044690
IF (XNORM + DXNORM .EQ. XNORM) GO TO 55	00044700
50 CONTINUE	00044710
IER = 130	00044720
C STORE THE SOLUTION	00044730
55 DO 60 JK = 1, N	00044740
B(JK, J) = WA(JK, N1)	00044750
60 CONTINUE	00044760
IF (IER .NE. 0) GO TO 9000	00044770
65 CONTINUE	00044780
GO TO 9005	00044790
9000 CONTINUE	00044800
CALL UERTST(IER, 6HLEQ2C)	00044810
9005 RETURN	00044820
END	00044830
C	00044840
C	00044850
C	00044860
SUBROUTINE LEQTC (A, N, IA, B, M, IB, IJOB, WA, IER)	00044870
C SPECIFICATIONS FOR ARGUMENTS	00044880
INTEGER N, IA, M, IB, IJOB, IER	00044890
COMPLEX*16 A(IA, N), B(IB, M)	00044900
DOUBLE PRECISION WA(N)	00044910
C SPECIFICATIONS FOR LOCAL VARIABLES	00044920
DOUBLE PRECISION P, Q, ZERO, ONE, T(2), RN, BIO	00044930
COMPLEX*16 SUM, TEMP	00044940
INTEGER I, J, JM1, IM1, K, IMAX, JF1, IW, N1	00044950

	EQUIVALENCE	(SUM,T(1))	00044960
	DATA	ZERO/0.000/,ONE/1.00/	00044970
C		INITIALIZATION	00044980
C		FIRST EXECUTABLE STATEMENT	00044990
	IER = 0		00045000
	IF (IJOB .EQ. 2) GO TO 75		00045010
	RN = N		00045020
C		FIND EQUILIBRATION FACTORS	00045030
	DO 10 I=1,N		00045040
	BIG = ZERO		00045050
	DO 5 J=1,N		00045060
	TEMP = A(I,J)		00045070
	P = CDABS(TEMP)		00045080
	IF (P .GT. BIG) BIG = P		00045090
5	CONTINUE		00045100
	IF (BIG .EQ. ZERO) GO TO 105		00045110
	WA(I) = ONE/BIG		00045120
10	CONTINUE		00045130
C		L-U DECOMPOSITION	00045140
	DO 70 J = 1,N		00045150
	JM1 = J-1		00045160
	IF (JM1 .LT. 1) GO TO 25		00045170
C		COMPUTE U(I,J), I=1,...,J-1	00045180
	DO 20 I=1,JM1		00045190
	SUM = A(I,J)		00045200
	IM1 = I-1		00045210
	IF (IM1 .LT. 1) GO TO 20		00045220
	DO 15 K=1,IM1		00045230
	SUM = SUM-A(I,K)*A(K,J)		00045240
15	CONTINUE		00045250
	A(I,J) = SUM		00045260
20	CONTINUE		00045270
25	P = ZERO		00045280
C		COMPUTE U(J,J) AND L(I,J), I=J+1,...,	00045290
	DO 45 I=J,N		00045300
	SUM = A(I,J)		00045310
	IF (JM1 .LT. 1) GO TO 40		00045320
	DO 35 K=1,JM1		00045330
	SUM = SUM-A(I,K)*A(K,J)		00045340
35	CONTINUE		00045350
	A(I,J) = SUM		00045360
40	Q = WA(I)*CDABS(SUM)		00045370
	IF (P .GE. Q) GO TO 45		00045380
	P = Q		00045390
	IMAX = I		00045400
45	CONTINUE		00045410
C		TEST FOR ALGORITHMIC SINGULARITY	00045420
	Q = RN+P		00045430
	IF (Q .EQ. RN) GO TO 105		00045440
	IF (J .EQ. IMAX) GO TO 60		00045450
C		INTERCHANGE ROWS J AND IMAX	00045460
	DO 50 K=1,N		00045470
	TEMP = A(IMAX,K)		00045480
	A(IMAX,K) = A(J,K)		00045490
	A(J,K) = TEMP		00045500
50	CONTINUE		00045510
	WA(IMAX) = WA(J)		00045520
60	WA(J) = IMAX		00045530
	JP1 = J+1		00045540
	IF (JP1 .GT. N) GO TO 70		00045550

C	TEMP = A(J,J)	DIVIDE BY PIVOT ELEMENT U(J,J)	00045560
	DO 65 I = JP1,N		00045570
	A(I,J) = A(I,J)/TEMP		00045580
65	CONTINUE		00045590
70	CONTINUE		00045600
75	IF (IJOB .EQ. 1) GO TO 9005		00045610
	DO 103 K = 1,M		00045620
C		SOLVE UX = Y FOR X	00045630
	IW = 0		00045640
	DO 90 I = 1,N		00045650
	IMAX = WA(I)		00045660
	SUM = B(IMAX,K)		00045670
	B(IMAX,K) = B(I,K)		00045680
	IF (IW .EQ. 0) GO TO 85		00045690
	IM1 = I-1		00045700
	DO 80 J = IW,IM1		00045710
	SUM = SUM - A(I,J)*B(J,K)		00045720
80	CONTINUE		00045730
	GO TO 88		00045740
85	IF (T(1) .NE. ZERO .OR. T(2) .NE. ZERO) IW = I		00045750
88	B(I,K) = SUM		00045760
90	CONTINUE		00045770
C		SOLVE LY = B FOR Y	00045780
	N1 = N+1		00045790
	DO 100 IW = 1,N		00045800
	I = N1-IW		00045810
	JP1 = I+1		00045820
	SUM = B(I,K)		00045830
	IF (JP1 .GT. N) GO TO 98		00045840
	DO 95 J = JP1,N		00045850
	SUM = SUM - A(I,J)*B(J,K)		00045860
95	CONTINUE		00045870
98	B(I,K) = SUM/A(I,I)		00045880
100	CONTINUE		00045890
103	CONTINUE		00045900
	GO TO 9005		00045910
C		ALGORITHMIC SINGULARITY	00045920
105	IER = 129		00045930
9000	CONTINUE		00045940
C		PRINT ERROR	00045950
	CALL UERTST(IER,6HLE9T1C)		00045960
9005	RETURN		00045970
	END		00045980
			00045990

SUPPLEMENTARY

INFORMATION



DEPARTMENT OF THE AIR FORCE
AIR FORCE WRIGHT AERONAUTICAL LABORATORIES (AFWL)
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433-6543

REPLY TO
ATTN OF: IMST (513/255-7466)

1 May 1987

SUBJECT: Correction to AFWAL Technical Reports, AFWAL-TR-86-3034
and 86-3035

ALL ADDRESSES

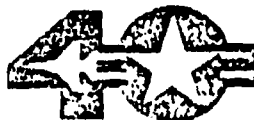
1. Please delete the second paragraph in the NOTICE page affixed to the inside cover of AFWAL-TR-86-3034, "Strength Analysis of Laminated and Metallic Plates Bolted Together by Many Fasteners" and AFWAL-TR-86-3035, "Design Guide for Bolted Joints in Composite Structures."
2. Please contact the undersigned if you have any questions regarding this letter.

G. Doben
G. DOBEN
Chief, Scientific & Tech Info Gp
Information Services Branch

cc: AFWAL/FIBRA
(V. Venkayya)

AD-B108123

UNITED STATES AIR FORCE



SEPTEMBER 18, 1947